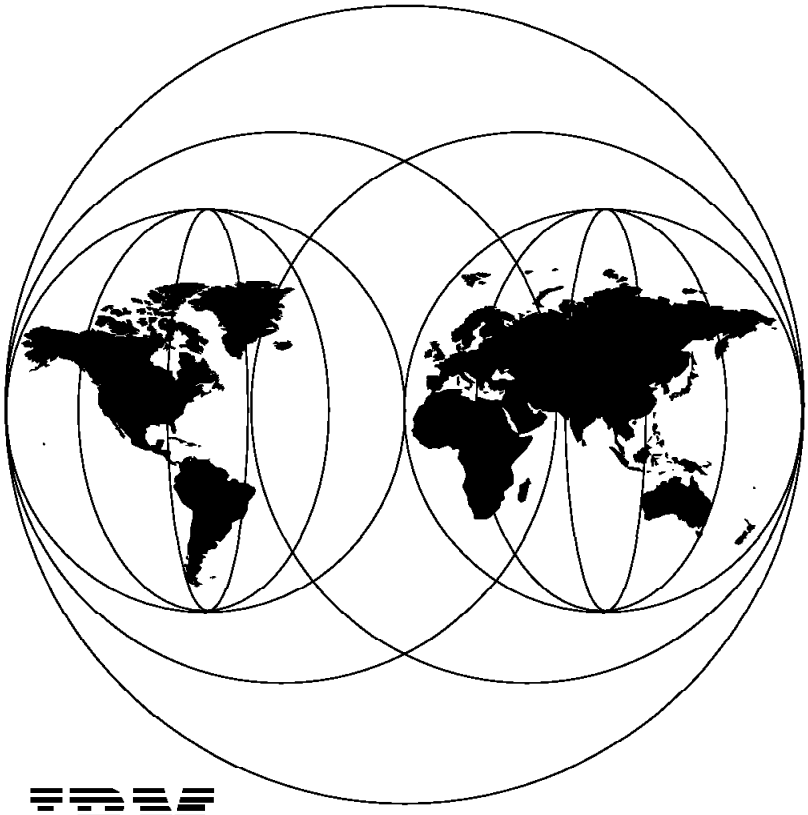
SG24-4680-01

OS/390 MVS Parallel Sysplex Capacity Planning

January 1998



IBM

International Technical Support Organization
Poughkeepsie Center



International Technical Support Organization

SG24-4680-01

OS/390 MVS Parallel Sysplex Capacity Planning

January 1998

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix F, “Special Notices” on page 301.

Second Edition (January 1998)

This edition applies to the following products, for use with OS/390 V2R4.

- CICS TS for OS/390 V1, 5655-147
- DB2 for OS/390 V5, 5695-DB2
- DFSMS/MVS V1.3, 5695-DF1
- IMS/ESA V6.1, 5655-158

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HYJ Mail Station P099
522 South Road
Poughkeepsie, New York 12601-5400

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1998. All rights reserved.**
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Figures ix

Tables xiii

Preface xv
The Team That Wrote This Redbook xvi
Comments Welcome xvii

Chapter 1. Introduction to Capacity Planning 1
1.1 Why Do I Need to Do Capacity Planning? 2
1.2 Capacity Planning Methods 4
1.2.1 Guidelines 4
1.2.2 Linear Projections 4
1.2.3 Analytic Methods 5
1.2.4 Discrete Methods 5
1.2.5 Benchmarks 5
1.3 Capacity Planning and CPU 8
1.3.1 From Cycle Time to M-value 8
1.3.2 OS/390 CPU Time Account and Capture Ratio 14
1.3.3 Balanced System Concept 20
1.3.4 Peak-to-Average Ratio (PAR) and Saturation Design Point (SDP) . . 21
1.3.5 MP Factor 29
1.3.6 SRM CPU Service Units 29
1.3.7 LPAR Capacity Planning 30
1.4 Capacity Planning and I/O 31
1.4.1 Relative I/O Content (RIOC) 31
1.4.2 SRM I/O Service Units 33
1.4.3 Relative Lock Content (RLKC) 34
1.5 Capacity Planning and Storage 35
1.5.1 Central Storage Capacity Planning 35
1.5.2 Parallel Sysplex Related Storage 35
1.5.3 Central and Expanded Storage Considerations 36
1.5.4 SRM Storage Service Units 38
1.6 Capacity Planning Activities 39
1.6.1 Decide Capacity Planning Method 40
1.6.2 Measure Current Workload 41
1.6.3 Check for a Balanced System 42
1.6.4 Predict the Future 42
1.6.5 Model and Interpret Data 43
1.6.6 Collect and Document the Capacity Planning Results 43
1.6.7 Communicate the Capacity Planning Results 44

Chapter 2. Capacity Planning in Parallel Sysplex 45
2.1 An Extension to the S/390 Architecture 46
2.1.1 Workload Management 48
2.1.2 Data Integrity and Buffer Pool Coherency Considerations 49
2.1.3 Locking in Parallel Sysplex 53
2.1.4 Parallel Sysplex HW Components 54
2.1.5 Cost Factors in Parallel Sysplex 62
2.2 DB2 Parallel Sysplex Data Sharing Considerations 63
2.2.1 DB2 Parallel Sysplex Data Sharing 64

© Copyright IBM Corp. 1998

2.2.2 No Expected I/O Growth for DB2 . . . 64
2.2.3 Parallel Sysplex CPU Cost for DB2 Data Sharing . . . 65
2.2.4 DB2 Locking Rate Calculations . . . 67
2.2.5 DB2 Buffer Pool Sizing in Parallel Sysplex . . . 70
2.3 IMS/DB Parallel Sysplex Data Sharing Considerations . . . 74
2.3.1 IMS/DB Parallel Sysplex Data Sharing . . . 75
2.3.2 Expected I/O Growth for IMS/DB . . . 76
2.3.3 Parallel Sysplex CPU Cost for IMS/DB Data Sharing . . . 77
2.3.4 IMS/DB Locking Rate Calculations . . . 78
2.3.5 IMS/DB Buffer Pools Sizing in Parallel Sysplex . . . 80
2.4 IMS DC Shared Message Queue (SMQ) . . . 82
2.5 CICS/VSAM RLS Data Sharing Considerations . . . 84
2.5.1 MRO Considerations . . . 86
2.5.2 VSAM RLS Parallel Sysplex Data Sharing . . . 87
2.5.3 Expected I/O Growth for RLS . . . 87
2.5.4 Parallel Sysplex CPU Cost for RLS Data Sharing . . . 88
2.5.5 VSAM RLS Locking Calculations . . . 89
2.5.6 VSAM Cache Usage and Sizing in a Parallel Sysplex . . . 91
2.6 CICS TS Logging Considerations . . . 93
2.6.1 DFHLOG Sizing . . . 96
2.6.2 DFHSHUNT Sizing . . . 97
2.6.3 Forward Recovery Logs Sizing . . . 97
2.6.4 User Journal and Autojournal Sizing . . . 97
2.6.5 Monitoring the Logger Environment . . . 98
2.7 CICS TS Temporary Storage . . . 98
2.7.1 CICS TS ITR Considerations . . . 100
2.7.2 TS Coupling Facility Sizing . . . 100

Chapter 3. OS/390 UNIX Services Sizing and Capacity Planning . . . 101
3.1 Introduction . . . 101
3.2 The UNIX Methodology . . . 102
3.2.1 The SPECint Benchmarks . . . 104
3.2.2 The TPC-C Benchmark . . . 104
3.2.3 The TPC-A Benchmark . . . 105
3.3 The Difference between TPC-C and Reality . . . 105
3.3.1 Comparison of Benchmarks . . . 106
3.4 Application Integration in a UNIX Environment . . . 107
3.4.1 Workload Management across a Cluster . . . 108
3.5 OS/390 UNIX Services: the OS/390 Perspective . . . 109
3.5.1 Creating New Processes . . . 110
3.5.2 Threads . . . 111
3.5.3 UNIX File System Usage . . . 112
3.5.4 Services Executed Using PC to OMVS Address Space . . . 112
3.5.5 Pathlength Considerations . . . 113
3.5.6 Memory Usage . . . 114
3.6 Sizing for the OS/390 UNIX Services Environment . . . 115
3.6.1 The Bridge Model . . . 116

Chapter 4. Using RMF Parallel Sysplex Reports . . . 117
4.1 RMF Parallel Sysplex Reports . . . 118
4.2 Coupling Facility Reports . . . 119
4.2.1 RMF CF Activity - Usage Summary . . . 120
4.2.2 RMF CF Activity - Structure Activity . . . 124
4.2.3 RMF CF Activity - Subchannel Activity . . . 126
4.2.4 Understanding CF Processing . . . 126

4.2.5 Summary of How to Read the RMF CF Activity Report 131
4.2.6 RMF CF Activity Spreadsheet 133
4.3 RMF Channel Activity Report 137
4.4 RMF XCF Activity Reports 138
4.5 RMF OMVS Kernel Activity Report 141

Chapter 5. A CF Processing Model 143
5.1 Introduction to Using the CF Processing Model 144
5.2 Description of the CF Processing Model 144
5.2.1 CF Request Processing 144
5.2.2 CF Times, Delays and Queues 147
5.2.3 A Model for the CF Request 148
5.2.4 Time Distributions and the k Factor 154
5.2.5 Verifying the Model 157
5.2.6 RMF Reports 157
5.3 Using the CF Processing Model 158

Chapter 6. Using CP2000 for Capacity Planning in a Parallel Sysplex 161
6.1 Capacity Planning with CP2000 162
6.1.1 Outline of CP2000 Usage 162
6.2 CP2000 Quick-Sizer Overview 164
6.2.1 Defining Your Environment to Quick-Sizer 165
6.2.2 Quick-Sizer Parallel Sysplex Migration 180
6.3 CF Access Rate Calculation 184
6.3.1 DB2 CF Access Rate Calculation 184
6.3.2 IMS/DB CF Access Rate Calculation 188
6.4 Other Parallel Sysplex Capacity Planning Issues 191
6.4.1 Workload and CPU Projections 191
6.4.2 Storage Projections 192
6.4.3 I/O Projections 194

Chapter 7. Parallel Sysplex Capacity Planning Case Studies 195
7.1 Case Study: DB2 Migration to a Data Sharing Environment 197
7.1.1 Quick-Sizer Default Input and Output 198
7.1.2 Quick-Sizer Detailed Input and Output 203
7.2 Case Study: Analyze Processor Speed 215
7.2.1 Case Description 216
7.2.2 Case Study Actions 216
7.2.3 Assumptions and Restrictions for the M/M/c Queuing Model 223
7.2.4 WLM Goal Mode 225
7.3 Case Study: CF Modelling 226
7.3.1 CF Model Case Study: Environment 226
7.3.2 CF Model Case Study: Questions 226
7.3.3 CF Model Spreadsheet 227
7.3.4 Defining the Input 228
7.3.5 Using the CF Spreadsheet Model 233
7.3.6 CF Model Case Study: Answers 238
7.3.7 CF Queuing Time Quick Graph 238
7.4 The OS/390 UNIX Services Sizer Tool 240
7.5 How to Use the Tool 240
7.5.1 Scenario One: Migrating an Existing UNIX Workload to S/390 240
7.5.2 Scenario Two: Adding an Additional UNIX Workload to S/390 245

Appendix A. Capacity Planning-Related Tools and Services 247

A.1 Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC) . . . 249
A.1.1 PCR Introduction and Background . . . 249
A.1.2 LSPR/PC . . . 251
A.2 LPAR/Capacity Estimator (LPAR/CE) . . . 253
A.2.1 Introduction to LPAR/CE . . . 253
A.2.2 LPAR/CE Implementation and Operation . . . 253
A.2.3 LPAR/CE Considerations for Parallel Sysplex . . . 256
A.3 RMF Spreadsheet Reporter . . . 259
A.4 Performance Management Services (PMOS) . . . 262
A.5 Enterprise Server Capacity Planning Oriented Tools Table . . . 267
A.6 Enterprise Server Capacity Planning Tools Catalogue . . . 271
A.6.1 Enterprise Server Capacity Planning and Other Tools Positioning . . . 272
A.6.2 Capacity Planning for CICS Cross Platform Migrations (CA17) . . . 274
A.6.3 CMOS Batch Analysis Tool (CBAT) . . . 274
A.6.4 CMOS Processor Utilization Tool (CD13) . . . 275
A.6.5 CP2000 (Capacity Planning 2000) . . . 276
A.6.6 Processor Capacity Reference (PCR) . . . 277
A.6.7 S/390 Parallel Sysplex Quick-Sizer (SPSSZR) . . . 278
A.6.8 CVRCAP (CICS/VSAM RLS Capacity Planning) . . . 278
A.6.9 DB2PARA (DB2 Parallel Sysplex Input for CP2000 Quick-Sizer) . . . 279
A.6.10 DASD Cache Analysis Tool (DCAT) . . . 279
A.6.11 DASD and Tape Magic/2 (DMAGIC2) . . . 280
A.6.12 IMSAFAID (IMS Affinity Aid) . . . 280
A.6.13 IMSSMQCP (IMS Shared Message Queue Capacity Planner) . . . 280
A.6.14 KGTV Tool, a CF Processing Model . . . 281
A.6.15 LPAR Capacity Estimator (LPAR/CE) . . . 281
A.6.16 Large Systems Performance Reference/PC (LSPR/PC) . . . 282
A.6.17 RLSLKSZ (RLS Lock Sizer) . . . 282
A.6.18 Capacity Effects of Software Migration (SOFTCAP) . . . 283
A.6.19 OS/390 UNIX Services Sizer Tool . . . 283
A.6.20 System Network Analysis Program/Simulation Host Overview Technique (SNAP/SHOT) . . . 284
A.6.21 Volume Mount Analyzer (VMA) . . . 284

Appendix B. Processor Storage Management and Extended Storage . . . 285

Appendix C. Providing Input to CP2000 . . . 287
C.1 SMF Parameters . . . 287
C.2 RMF Monitor I . . . 288
C.3 RMF Postprocessor . . . 288
C.4 RMF Monitor III . . . 290
C.5 CP2KEXTR . . . 290
C.6 ICS/IPS Parameters . . . 291
C.7 Configuration Data . . . 291
C.8 Knowledge of Workloads . . . 291

Appendix D. M-value Table (December 1997) . . . 293

Appendix E. Queueing Theory Concepts . . . 297
E.1 A Taxi Journey Example . . . 298
E.2 Relation between Mean Square, Variance and Coefficient of Variation . . . 299
E.3 Application to Delayed Synchronous Immediate Requests . . . 299
E.4 What about Queueing . . . 300

Appendix F. Special Notices 301

Appendix G. Related Publications 305
G.1 International Technical Support Organization Publications 305
G.2 Redbooks on CD-ROMs 305
G.3 Other Publications 306
G.3.1 CPSTOOLS Packages and Publications 306
G.3.2 MKTTOOLS Packages and Publications 306
G.3.3 IBM Standard Publications 306
G.3.4 Washington Systems Center Flashes 307
G.3.5 Other Books and Articles 308
G.4 How to Reach CP2000, CPAIP, LPAR/CE, LSPR/PC, PCR and Quick-Sizer 308

How to Get ITSO Redbooks 309
How IBM Employees Can Get ITSO Redbooks 309
How Customers Can Get ITSO Redbooks 310
IBM Redbook Order Form 311

Glossary 313

List of Abbreviations 327

Index 331

ITSO Redbook Evaluation 333

Figures

1. A Stressed CEC . . . 2
2. Capacity Planning Methods Positioning . . . 7
3. Captured and Non-Captured CPU Utilization . . . 15
4. CPU Utilization after Applying CR . . . 17
5. CP Load Distribution - Light Load . . . 23
6. CP Load Distribution - Maximum Load . . . 24
7. CP Load Distribution - Overload . . . 25
8. CP Response Time versus Utilization for Various Number of CPs . . . 26
9. Response Time over Utilization As an Effect of CP Speed . . . 28
10. Capacity Planning Activities - Step-by-Step . . . 40
11. Scalability in the Parallel Sysplex . . . 47
12. Multisystem Data-Sharing . . . 50
13. Multisystem Data Sharing in Parallel Sysplex . . . 51
14. DB2 Buffer Pools, Single OS/390 versus Parallel Sysplex . . . 70
15. IMS/DB Buffer Pools, Single OS/390 versus Parallel Sysplex . . . 80
16. CICS Configuration Using Local Files versus CICS/VSAM RLS . . . 86
17. CICS Cache Set, Single OS/390 versus Parallel Sysplex . . . 92
18. CICS TS Logging Environment . . . 94
19. CICS TS Temporary Storage . . . 99
20. Typical UNIX/PC Benchmarks . . . 103
21. The Capacity Curve . . . 106
22. Application Integration in the UNIX Environment . . . 107
23. Workload Balancing across a Cluster . . . 108
24. Schematic Overview of Fork() . . . 110
25. Schematic Overview of Program Call . . . 113
26. The Relative Capacity Problem . . . 115
27. RMF Data Collection and Reporting for CF Resources . . . 118
28. RMF CF Activity Report - Usage Summary . . . 123
29. RMF CF Activity Report - Structure Activity . . . 124
30. RMF CF Activity Report - Subchannel Activity . . . 126
31. CF Requests and Delay Types . . . 127
32. RMF CF Activity - Structure and Subchannel Activity Sections . . . 129
33. RMF CF Activity Spreadsheet - Structure . . . 133
34. RMF CF Activity Spreadsheet - Structure MMDB2_GBP: . . . 134
35. RMF CF Activity Report - Subchannel Activity . . . 135
36. RMF CF Activity Spreadsheet - Engine Utilization for Synchronous Requests . . . 136
37. RMF CF Activity Spreadsheet - Link Buffer Utilization by System . . . 137
38. RMF Channel Activity Report . . . 138
39. RMF XCF Activity Report - Usage by System . . . 139
40. RMF XCF Activity Report - Usage by Member . . . 139
41. RMF XCF Activity Report - Path Statistics . . . 140
42. Sample RMF Report Output for OMVS Kernel Activity . . . 141
43. Some of the Components Participating in a CF Request . . . 145
44. CF Requests and Delay Types . . . 147
45. A Model for the CF Request . . . 148
46. Typical Inter-arrival Time Distributions . . . 154
47. Two Separate Service Time Distributions . . . 155
48. Combined Service Time Distribution (Unlimited CF Capacity) . . . 156
49. Combined Service Time Distribution (Limited CF Capacity) . . . 157
50. Relationship to RMF Reporting . . . 158

© Copyright IBM Corp. 1998

51. Overview of CP2000 . . . 162
52. Capacity Planning Roadmap . . . 163
53. Quick-Sizer Structure . . . 165
54. Quick-Sizer Input Window . . . 166
55. Quick-Sizer Optional Input Window . . . 169
56. Quick-Sizer Output . . . 171
57. Quick-Sizer Service Time Analysis . . . 176
58. Quick-Sizer Coupling Facility Structure Display . . . 177
59. Quick-Sizer Defaults . . . 178
60. Parallel Sysplex Workload Migration - Overview . . . 181
61. Quick-Sizer Defaults for DB2 . . . 186
62. Quick-Sizer Defaults for IMS . . . 189
63. The Range of Storage Sizes over the M-Value . . . 193
64. Quick-Sizer Input Window . . . 198
65. Quick-Sizer Optional Input Window 2 . . . 200
66. Quick-Sizer Output Window . . . 202
67. Quick-Sizer Defaults Window (Scrolled Left) . . . 203
68. DB2PM Accounting Report (Lock Information) . . . 204
69. DB2PM Statistics Report (Lock Information) . . . 205
70. Quick-Sizer Defaults Window (Scrolled Right) . . . 207
71. DB2PM Accounting Report (Buffer Pool Information) . . . 208
72. DB2PM Statistics Report (Buffer Pool Information) . . . 209
73. Quick-Sizer Output Window 2 . . . 212
74. Quick-Sizer Display CF Window . . . 213
75. Quick-Sizer Processor Utilization Graph . . . 214
76. Workload Priority Influence on CPU Response Time: an Example . . . 215
77. CP2000 Comparing Processors Panel . . . 217
78. CP2000 Workload CPU Response Time Model for System Image . . . 218
79. CP2000 Analyze Processor Speed . . . 219
80. CP2000 Processor Utilization Graph . . . 219
81. CP2000 Processor Service Time Graph . . . 220
82. CP2000 Processor Response Time per Workload per CEC Graph . . . 220
83. CP2000 Single CP Analysis Graph . . . 221
84. CP2000 Processor Response Time Graph versus Processor Utilization Graph . . . 222
85. CP2000 Processor Response Time versus Transaction Rate Graph . . . 223
86. Case Study Environment . . . 226
87. Input from RMF Subchannel Activity Section . . . 229
88. Input from RMF Usage Summary Section . . . 233
89. Case Study Results - Total Processing Time Components . . . 236
90. Case Study Results - Service and Processing Time Changes . . . 237
91. Case Study Results - CF Queuing Time . . . 239
92. OS/390 UNIX Services Sizer Logo Screen . . . 241
93. OS/390 UNIX Services Sizer Primary Panel . . . 241
94. OS/390 UNIX Services Sizer Input Details . . . 242
95. OS/390 UNIX Services Sizer Computation . . . 243
96. OS/390 UNIX Services Sizer OS/390 Result . . . 243
97. OS/390 UNIX Services Sizer MVS Result . . . 244
98. OS/390 UNIX Services Sizer Processor Selection . . . 245
99. OS/390 UNIX Services Sizer 2nd Application . . . 245
100. OS/390 UNIX Services Sizer 2nd Example . . . 246
101. OS/390 UNIX Services Sizer Primary Panel . . . 246
102. PCR Processor Table Window . . . 251
103. LPAR/CE Input and Summary Panel . . . 254
104. LPAR/CE Real CP Resource Usage Panel . . . 255

105. LPAR/CE Partition Capacity Panel 255
106. LPAR/CE Analysis Summary 256
107. RMF Spreadsheet Reporter - Sample Cache Performance Overview in Spreadsheet 259
108. Average Requests by System for Each CF 263
109. Positioning of Capacity Planning Tools 273
110. Storage Pages and CS/ES Storage 285

Tables

1. Typical Application Capture Ratios for Different Workloads (1997) 19
2. Typical Non-Sysplex RIOCs for Different Workloads (1997) 33
3. IBM CECs Participating in Parallel Sysplex 55
4. CF Structures for IBM Products 57
5. Multimode CF Link Effect on ITR 60
6. Reduction in Sender ITR/km CF Link (Because of Synchronous CF Access) 61
7. Relative Importance to Increase in Resource Requirement 62
8. Coefficient (E) for Enablement Cost of IMS DB, DB2 and CICS/VSAM RLS Data Sharing 66
9. Parallel Sysplex DB2 Buffer Pool Sizing Summary 74
10. Comparison of Different Benchmarks 107
11. Summary of Delay Reasons and Affected Counters in RMF CF Activity Report 130
12. Coupling Facility Access Service Times for Lock Requests 132
13. Examples of CEC CP Consumption Per Synchronous CF Requests . . 159
14. Examples of Synchronous CF Request Response Time Improvements 159
15. Input for Transaction/Database Manager Selection 170
16. Access Rate Input for Transaction/Database Managers 179
17. Input Table (Worksheet Input_1) 229
18. Results (Worksheet Input_1) 229
19. General and CEC Specific Input Data (Worksheet Input_2) 230
20. CF Input Data (Worksheet Input_2) 231
21. Link Input Data (Worksheet Input_2) 231
22. Model Output and Scenarios (Worksheet Model) 234
23. RMF Spreadsheet Reporter - Available Spreadsheet Macros 260
24. S/390 Capacity Planning Tools - Areas Covered 268
25. M-values (December 1997) 293

© Copyright IBM Corp. 1998

Preface

Capacity planning in the Parallel Sysplex environment must address specific transition and growth considerations.

Opportunities exist for balancing your workload across systems in a Parallel Sysplex. Resources such as systems, Coupling Facilities and links belonging to a Parallel Sysplex have a finite capacity which, when exhausted, may slow down or disrupt service to users. You can avoid these conditions and significantly increase the productivity of users by careful capacity planning.

This redbook provides an overview of capacity planning. It defines the process and the vocabulary, and provides detailed explanations about how to do capacity planning in the Parallel Sysplex environment. Related IBM tools that help in this process are also discussed.

Capacity planning considerations for OS/390 V2R4 and related subsystems such as the DB2, IMS/DB and CICS/VSAM RLS data sharing environments are included. For the DB2 environment, a methodology is outlined for estimating the CF locking rates and for obtaining locking information based on DB2PM reports from your current system. A detailed description of RMF reporting for a Parallel Sysplex is provided.

Case studies are included for Quick-Sizer modelling of DB2 and IMS/DB data sharing. These case studies are constructed so that you can try them while reading the redbook. Additionally, information and case studies are included for UNIX capacity planning in a S/390 environment. A discussion and case study are included for a CF processing model, which enables you to evaluate proposed changes to your Parallel Sysplex configuration.

This redbook is intended for IBM large systems customers and IBM systems professionals, such as availability services specialists. Some knowledge of Parallel Sysplex and capacity planning concepts is assumed.

© Copyright IBM Corp. 1998

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization Poughkeepsie Center. These projects was designed, managed and partly documented by:

Henrik Thorsen International Technical Support Organization, Poughkeepsie

The authors of the original document and this edition in alphabetical order are:

Paola Bari IBM USA
Per Fremstad IBM Norway
Keith George IBM UK
Raimo Korhonen CompMeas Consulting Ltd Finland
Christian Matthys IBM France
Hans-Dieter Mertiens IBM Germany
Alfred Prinz IBM Germany
Alvaro Salla IBM Brazil
Norbert Schorm IBM Germany
Jan van Kempen IBM Netherlands
Robert Vaupel IBM Germany

Thanks to the following people for their invaluable contributions to this project:

Justo Abejon IBM France
Steve Anania IBM USA
Linda August IBM USA
Svend Erik Bach IBM Denmark
Evanne Bernardo IBM USA
Runey Bonilha Ribeiro IBM Brazil
Page Borchetta IBM USA
Fulvio Capogrosso IBM Italy
Dave Clitherow IBM UK
Rich Conway IBM USA
Roy Cornford IBM UK
Noshir Dhondy IBM USA
John Fitch IBM USA
Gretchen Frye IBM USA
Bob Gutjahr IBM USA
Audrey Helffrich IBM USA
Gary King IBM USA
Gopal Krishnan IBM USA
Ravi Kumar IBM USA
Rick Lebsack IBM USA
Paul Lekkas IBM Germany
Bill Melnichuk IBM Canada
Franco Meroni IBM Italy
Jean Michel-Levy IBM France
Dennis Moore IBM USA
Richard Pape IBM Malaysia
Michel Plouin IBM France
Alex Patterson IBM USA
Dave Raften IBM USA
Bernice Riley IBM USA
Jim Shaw IBM USA
Joe Shelak IBM USA
Akira Shibamiya IBM USA
Hugh Smith IBM USA
Joe Temple IBM USA
Michael Teuffel IBM Germany
Stan Washington IBM USA
Ray Wicks IBM USA
Mark Wisniewski IBM USA

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “ITSO Redbook Evaluation” on page 333 to the fax number shown on the form.
- Use the electronic evaluation form found on the Redbooks Web sites:

 For Internet users `http://www.redbooks.ibm.com`
 For IBM Intranet users `http://w3.itso.ibm.com`

- Send us a note at the following address:

 `redbook@vnet.ibm.com`

Chapter 1. Introduction to Capacity Planning

At a Glance

This chapter discusses the capacity planning process, with some emphasis on Parallel Sysplex. It offers different capacity planning methods and highlights the benefits of careful capacity planning.

A high-level overview is given of several tools and methods that are useful when doing capacity planning.

Recommended Reading:

- *Probability, Statistics, and Queueing Theory with Computer Science Applications*, A.O. Allen, Academic Press
- *Large Systems Performance Reference (LSPR)*, SC28-1187.

© Copyright IBM Corp. 1998

planning studies.

1.1 Why Do I Need to Do Capacity Planning?

When making a journey into an area where there are few opportunities to stop for fuel, it is a good idea to carry the needed fuel, plan to refuel, and to know how far one can go with the fuel that the vehicle was designed to hold. No one wants to be out of gas.

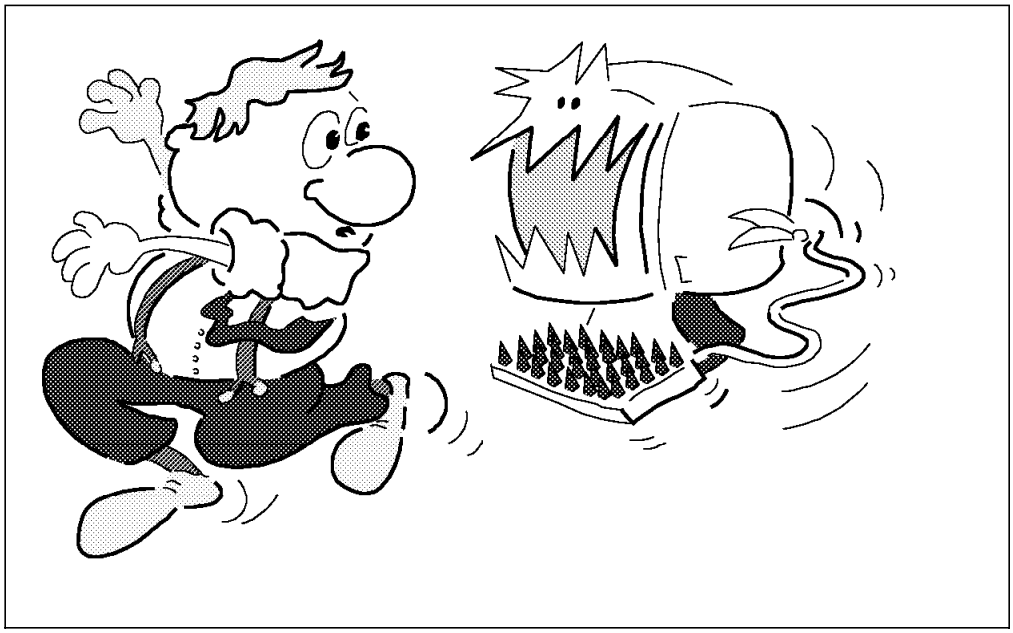


Figure 1. A Stressed CEC

Out of gas?

A Central Electronics Complex (CEC) usually does not stop when it reaches the out-of-gas condition, even if it encounters dead locks, loops, and other conditions that bring it into a wait state. However, all CECs have a certain capacity and can run out of gas when they are given more work to process than the processor can execute within the expected time period. We can avoid this out-of-gas condition and significantly increase the productivity of the users with proper capacity planning. You must monitor and look at the resources early, when everything is still working.

Doing a capacity planning study means to define a process for planning for sufficient CEC capacity in a cost-effective manner to meet the service needs of your users.
To do capacity planning, the following questions should be answered:

- Which Information Technology (IT) resources are currently in use? (CPU, CF, devices, controllers, channels, coupling links, storage, and so forth.)
- Which parts of the workloads consume most of the resources?
- What is the expected growth rate for particular workloads?
- When are the demands on current resources going to impact delivered service levels?

These questions should be answered for a *balanced* system as a starting point.

Benefits of Capacity Planning

Resources, in a balanced system, are not independent variables in the determination of system performance. The interrelationships of processing power, I/O capability, processor storage size, and networks determine system performance. Capacity can then be determined based on the performance objectives.

A balanced system is a system where all the resources are dimensioned toward each other in order to optimize the system's capacity and get the best performance and throughput in accordance with the goals.

On average, there should be no single bottleneck in a balanced system. If there is a single bottleneck, then it should be the most expensive one to eliminate, *and* all other bottlenecks have almost been eliminated. It is extremely difficult to do capacity planning if you have a major bottleneck in your system. An example of this is a CPU-constrained system.

If you achieve a "balanced system," (for details refer to *Balanced Systems and Capacity Planning*, GG22-9299), you will enjoy the following benefits during and after the capacity planning sessions or tasks:

- Optimizing your financial investments
- Increasing IT productivity
- Increasing communication between users and system service providers
- More transparency for management by establishing service level agreements
- Avoiding performance problems that will seriously impact your business in the future

Capacity Planning and Parallel Sysplex

Historically, we measured the following three main parts of a balanced system for capacity planning:

- CPU
- Storage
 - Central
 - Expanded
- I/O
 - Channels
 - Controllers (storage directors, caches, and NVS)
 - Devices

In a Parallel Sysplex environment, the following resources are monitored:

- CPU
- Storage
 - Central
 - Expanded
- I/O
 - Channels
 - Controllers (storage directors, caches, and NVS)
 - Devices
- CF
 - Links
 - CPU
 - Storage

When operating as a Parallel Sysplex, or when planning to migrate to Parallel Sysplex, the impact of Parallel Sysplex on these resources needs to be assessed.

Capacity Planning and CF

Parallel Sysplex introduces the Coupling Facility (CF) as an additional resource, formed by:

- Links
- CPU
- Storage

The CF is a key resource in a Parallel Sysplex configuration. It is a kind of *intelligent and shared memory* that allows the sharing of data. It enables an infrastructure suitable for workload distribution and resource balancing across multiple OS/390 images. This provides a single image view of the Parallel Sysplex and preserves integrity of shared data.

The CF is accessed via OS/390 services executed either in synchronous or asynchronous mode. These services were introduced in MVS/ESA V5. In a capacity planning study it is important to size the resources belonging to the CF to get a *balanced* Parallel Sysplex environment.

1.2 Capacity Planning Methods

Depending on available time, cost, skill, and level of detail and accuracy, you as capacity planner can choose the method that best meets your expectations. We will outline and assess several different methods to provide you with the ability to choose a suitable method.

1.2.1 Guidelines

A guideline is somewhere between a guess and a rule of thumb, and is simple to execute. Simple, informal projections are often done without documentation and may be based on experience and knowledge of the system. An example of a guideline is when historical workload utilization is plotted and a straight line is used to predict the future. Using this approach, additional CP resources are added or the workload split when a guideline (expressed as a rule of thumb) is violated. Pen, paper, and calculator are often the tools used to create guidelines for Capacity Planning methods. Accuracy cannot usually be predicted with such methods.

1.2.2 Linear Projections

A linear projection is anything from simple diagrams made with spreadsheets to more sophisticated methods. LPAR Capacity Estimator, CP2000 Quick-Sizer, and Processor Capacity Reference (PCR), (based on LSPR data) are examples of tools that can help you estimate various configurations and workloads using a spreadsheet-like approach.

For details, refer to A.1, “Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)” on page 249, A.2, “LPAR/Capacity Estimator (LPAR/CE)” on page 253, and 6.2, “CP2000 Quick-Sizer Overview” on page 164.

The accuracy bandwidth of projection tools is about 20% for utilization.

1.2.3 Analytic Methods

Based on mathematical methods such as the queueing theory, analytic methods can provide insight into forecasting CPU utilization, response time evaluation, capture ratios, effect of buffering, effect of queueing, and so forth. Based on measured data from today′s environment, various configurations and growth scenarios are studied and documented. CP2000, Performance Reporter for OS/390 (EPDM) and BEST/1 are a subset of many IBM and non-IBM tools that fall into this category.

Generally, analytic tools have an accuracy bandwidth between 8% to 15% for utilization. This depends on the scenario, level of detail, and provided data.

1.2.4 Discrete Methods

A discrete method is an application of discrete *simulation*. Unlike analytic simulation, discrete simulation is *not* based on mathematical formulas. Using discrete capacity planning methods, dissimilar workloads and their effect on each other are modelled. An example is the effect of combining batch and interactive workloads on the same OS/390 system.

A discrete simulator simulates events that take place in a system, such as the input of a message to the system, its arrival at the CPU, the instructions required to process it, polling the terminal, and sending the response. All these events are simulated, tabulated, and reported by the discrete simulation model. Sometimes a discrete simulation requires more detailed input data than an analytic simulation. SNAP/SHOT is an example of a simulation tool and service.

Discrete simulation tools usually have an accuracy bandwidth of 5% to 10% for utilization. However, the accuracy may be +/- 30% for response time. This depends on the scenario complexity and level of detail.

1.2.5 Benchmarks

Central Electronics Complex (CEC)[1] power is measured via benchmarks. Benchmarking means running a real workload in a particular hardware configuration to measure the CPU power. It is often more effective than the previous methods, but also more expensive and time consuming.

Two Types of Benchmarks

Benchmarks are divided into two types:

- Other benchmarks

 You can use evaluations from benchmark studies conducted by consulting companies, for example IBM.

- Your own benchmarks

 You run your own workload on a specific hardware configuration, which resembles a part or all of the anticipated configurations.

Unlike the analytic and discrete methods, the benchmark requires executable code and installed hardware to predict how a certain combination of resources will perform in a specific situation. This allows analysis of many different configurations and workload possibilities, often with just a change of a parameter.

[1] In some IBM literature, a CEC is also referred to as Central Processor Complex (CPC)

Benchmarks allow measurements of things such as locking, internal serialization, I/O (DASD, control units, strings), storage, and release-to-release software migrations.

Other Benchmarks

In many situations, to set up and complete a benchmark is very time consuming, expensive, and sometimes not possible at all. Large System Performance Reference (LSPR) is an example of benchmarks conducted by IBM. Both IBM and OEM CECs are evaluated by IBM. Refer to 1.3.1, "From Cycle Time to M-value" on page 8 and A.1.2, "LSPR/PC" on page 251 for further details on LSPR. The results derived from benchmark runs are often used in capacity planning. CP2000 and SNAP/SHOT are examples that use performance figures from LSPR expressed as internal throughput rates also known as ITRs. For a definition of ITRs, refer to 1.3.1.1, "CPU Time Measurement Methods" on page 9.

Consultant MIPs

Some consulting companies that offer CPU capacity information do not measure CECs; hence their evaluation is based on vendor claims or calculated as extrapolations, such as the Gartner MIPS.

TPC-C Benchmark

This type of benchmark is predominantly used in the UNIX and PC-based world. It is defined by the Transaction Processing Council (TPC) in a very detailed manner and the results are certified by the TPC. As we will explain later, we consider this to be an artificial benchmark with well-known drawbacks that is unsuitable for the S/390 environment. In fact, there are no S/390 TPC-C figures available. Nevertheless, the OS/390 UNIX Services Sizer uses rules of thumb to estimate the capacity required to move from UNIX to OS/390 UNIX Services based on UNIX TPC-C numbers. For a more detailed discussion, see Chapter 3, "OS/390 UNIX Services Sizing and Capacity Planning" on page 101.

SPEC Benchmark

This type of benchmark is again predominantly used in the UNIX and PC-based world. Its aim is to measure the speed of the processor, and its memory subsystem including all levels of cache. Again, we think this is not the way to classify our environments as it focuses on only a small piece of the total environment. For a more detailed discussion, see Chapter 3, "OS/390 UNIX Services Sizing and Capacity Planning" on page 101.

Your Own Benchmark

The validation of a Parallel Sysplex configuration can also be done using actual workloads and data. IBM provides services to allow Parallel Sysplex benchmarks on Parallel Sysplex configurations. An example of such a service is Solution Validation Services (SVS).

Benchmark Accuracy

Benchmark reachable accuracy is usually very high; it has an accuracy of 95% to 100%, depending on the scenario complexity.

Benchmark Services Available

IBM provides benchmark services if you are interested in evaluating your application environment in a Parallel Sysplex. Examples include:

SVS (Solution Validation Services) SVS is used to run your application at production volumes to validate capacity and find and fix problems before going to production. SVS provides expertise, equipment, and tools to assist in all the steps required to stress test and performance test your application. The contact point for this service is either through the IBM marketing representative or through a note to HQVMIC1(GLASS).

EMEA The S/390 Parallel Sysplex System Center (PSSC) in Montpellier (France) is available to run benchmarks for installations that need to evaluate and stress their applications in a Parallel Sysplex environment. The contact point is through the marketing representative or via a note to MOPPROFS(JUSTCALL).

Other Countries Certain countries are also doing their own benchmark studies. To obtain more information about this capability, contact your IBM marketing representative.

Figure 2 illustrates the positioning of the above-mentioned capacity planning methods. The figure merely shows that there is a relationship between “cost” and level of detail or accuracy. In the figure, all methods are shown rising from the x-axis. Methods can, in reality, have an accuracy of as low as 0%. This is, for example, the case when you run a very detailed benchmark while neglecting a bottleneck in the system. Figure 109 on page 273 provides an overview of several capacity planning tools mapped to these methods.

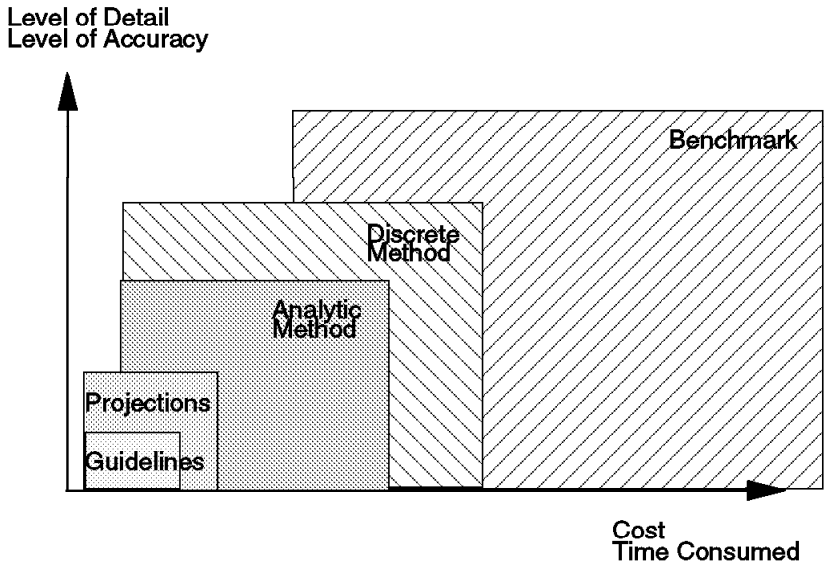


Figure 2. Capacity Planning Methods Positioning

1.3 Capacity Planning and CPU

This section contains a description of capacity planning terminology related to the CPU resource. We discuss what kind of data is involved in a capacity planning study and where you can find this data.

1.3.1 From Cycle Time to M-value

The fastest CEC for your workload is one that executes your workload in the least CPU time. CPU performance to execute a certain function can be expressed as the CPU time required to execute that function.

CPU Time Definition

The CPU time is derived using the following formula, where *Cycle time* is measured in nanoseconds:

$$CPU\ time = Path\ length \times Cycles/Instruction \times Cycle\ time$$

where:

- *Path length* is the set of instructions required to execute the workload. It relates to the computer architecture, but also to compiler efficiency.
- *Cycles/instruction* identifies how many CPU cycles on average are required to complete an instruction; it is a consequence of the hardware design, including the microcode.

 Cycles/instruction is often related to the *complexity* of the instruction. The number of cycles to execute a general instruction such as “CR” (compare) is less than that of the control instruction “MVPG” (move-page).
- *Cycle time* is the machine speed and is dependent on the technology.

MHz

The inverse value of the cycle time is the so-called oscillating frequency, or MHz, value. MHz is used especially with PCs and in the UNIX world.

No Universal Best Solution to Measure CEC Power

There are many ways to measure CEC performance, but none are absolutely perfect and there is *no universal performance scale*. If you only use *cycle time*, you are discarding the other important factor in your evaluation, path length. If you only use Millions of Instructions Per Second (*MIPS*), where MIPS is calculated with the following formula:

$$MIPS = \frac{1000}{(cycles/instruction \times cycle\ time)}$$

you can evaluate only CECs with the *same set of instructions* and with the *same compiler.*

Meaningless Indication of Processor Speed

The instruction mix per workload may vary greatly. The speed of different instructions may also vary among CECs. MIPS is therefore a misleading unit of measure when comparing the power of CECs.

CPU time is a good measure to evaluate CPU power. When comparing different CECs, you need to benchmark your workload in a non-constrained environment.

The MHz Battle

In the UNIX and PC world there is an ongoing MHz battle. Every year the speed of microprocessors increases. Other improvements, such as architecture, base technology, design changes happen only from time to time. Often the gain in throughput is related to increased MHz of the microprocessors. In the world of S/390 there are ongoing improvements in microprocessor speed, as well as to the hardware and software *architecture*. Situations may arise where throughput increases even though the basic speed of the processor does not. So MHz as a stand-alone metric is a not a good measure for processor comparisons.

MIPS Today Are Really RIPS

Another possibility is to use Relative Instructions Per Second (RIPS) to compare CECs. Here, you mention the particular CEC that is the reference point, and its RIPS value. Further, you have to describe the workload mix and operating system used. The starting RIPS value is selected arbitrarily (for example, close to the MIPS value claimed by you or a consultant) *if the RIPS values for other CECs are scaled correctly.* This should be based on LSPR measurements because it is a reliable source for comparisons, widely accepted inside and outside IBM. One advantage of using RIPS is that the path lengths for online transactions are easily tackled because they fit into this environment as well.

A practical implementation of RIPS used by some sources is shown in the following formula:

$$MIPS \approx 0.9 \times RPP$$

The Relative Processing Power (RPP) is discussed in the next section.

1.3.1.1 CPU Time Measurement Methods

In benchmarking your workload, methods for CPU measurement exist that give you higher accuracy than MIPS. Let us discuss some of these methods.

LSPR Large Systems Performance Reference

LSPR (Large Systems Performance Reference) is a set of IBM-conducted benchmarks. LSPR data consists of a comprehensive set of workloads, each representing a type of production environment. LSPR results are based on measurements and analysis across many contemporary System 390 architecture CECs, both IBM and IBM compatible. Workloads include various batch and online environments.

Each LSPR workload is a unique benchmark. Each has been developed to provide a reasonable representation of some major type of data processing activity, such as commercial batch, CICS, DB2, IMS, TSO, or VM/CMS. We usually expect LSPR numbers to be able to estimate the relative capacity of S/390 processors for a given workload to within 5%. This is far more accurate than TPC-C is likely to be between UNIX processors.

The benchmark figures published in LSPR are taken in a non-data-sharing environment. These benchmark workloads have also been run in data-sharing mode. Note, however, that running in data-sharing mode implies the use of specific models of CF, and running in a data-sharing group with two or more members. In other words, a large number of possible combinations have to be measured. We use the LSPR data and data obtained from specific data-sharing environments to project capacity requirements in any given data-sharing environment.

LSPR Introduction and Background: CECs should not be compared by using a single number. When comparing two CECs, it is important to know what kind of workload is used in the comparisons. Different CECs have different designs. CEC designs differ, for example, in the bandwidth of data paths between storage and CPs. CEC designs may also differ in size, speed, and layout of high-speed buffers. In addition, certain CECs are optimized for certain workloads by using special fast instructions implemented in hardware or microcode. Different workloads use different instruction mixes. Consequently, results of CEC comparisons are *dependent on the workload*. Other factors include the operating system, including level and maintenance level. Because the different operating systems, and the different levels of each, may use a different instruction mix, the results of the comparisons may be different. Also, factors such as CPU load and possible constraints in the configurations may affect the comparisons.

There are several ways to establish a capacity expectation for a new CEC. For example, a correct customized individual benchmark will produce the most accurate capacity data; however, benchmarks are expensive. A reliable alternative is IBM′s Large System Performance (LSPR) method. This method aims to provide comprehensive S/390 CEC capacity and is based on *workload sensitive measurements.* The focus is on CEC capacity, assuming sufficient external resources to prevent any constraints.

LSPR benchmarks are laboratory-controlled workload-sensitive tests that try to achieve the accuracy of customized benchmarks. If, for example, you use MIPS tables, your potential errors can be significant. Studies of LSPR data show that single-number MIPS entries in CEC tables can misstate relative CEC power by up to 100%.

LSPR should be viewed as a tool that can minimize the potential for misstating relative CEC capacity. If, however, MIPS (or RIPS) is the CEC capacity metric of your choice, you should use LSPR ITR data to calculate capacity for a potential new CEC. If you assume that your current installed CEC represents some MIPS to you, you can use the following formula to compare the capacity of two CECs:

$$MIPS_{New\ CEC} = MIPS_{Old\ CEC} \times \frac{(ITRR_{New\ CEC})}{(ITRR_{Old\ CEC})}$$

The goal of the LSPR is to offer reliable relative capacity information, which takes into account CEC design sensitivities to workload type. LSPR measurements are done in unconstrained environments at well-defined levels of CPU load, using well-defined workloads at well-defined operating system levels. The results are “clean” CEC-to-CEC comparisons in specified environments.

IBM distributes LSPR data in both the PC-based tools PCR and LSPR/PC. These tools are available for IBM representatives. For more information, refer to A.1, “Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)” on page 249.

The LSPR results are taken as the base of comparisons by many parties who publish comparative figures about CEC powers. Other providers of performance numbers include analysts such as:

- Annex
- Gartner Group, Inc.
- International Data Corporation, IDC

- Meta Group, Inc.
- Xephon

Often the numbers provided are based on vendor claims, *not on actual measurements.*

The unit of measure provided by LSPR is Internal Throughput Rate (ITR). ITR takes into account *all variables* that can exist when CECs are evaluated.

ITR Definition

$$ITR = \frac{Unit\ of\ works}{CPU\ time}$$

Unit of works can be expressed, for example, as jobs, job steps, or transactions. ITR is a function of:

- CP speed
- Microcode
- SCP (version, level and specific setup)
- Type and size of the transaction

ITRR

CECs can be compared using *ITR ratios* (ITRRs), where the ITR_{Value} is scaled to a base value ITR_{Base}.

$$ITRR = \frac{ITR_{Value}}{ITR_{Base}}$$

RPP

LSPR measurements can also be expressed as RPP (Relative Processing Power). When comparing different CECs, RPP is commonly used and accepted as a unit of measurement. RPP values are based on specific operating conditions and can be expressed for different types and versions of operating systems. One CEC has a base value, and others are scaled accordingly. The RPP values are calculated as a *weighted average* of particular well-defined LSPR workload ITRRs. An example of such a weighted average is:

- 12.5% CB84 (old commercial batch workload)
- 12.5% CBW2 (new, heavier commercial batch workload)
- 25.0% TSO
- 25.0% CICS
- 12.5% DB2/IMS
- 12.5% IMS or IMS/DS

The ITRRs are measured at 90% CPU utilization in an unconstrained environment. You may compose your own RPP value to represent your personal workload profile. Never use RPP values to compare CECs using different operating systems.

Relationship between ITR and RPP

A general definition of RPP is illustrated in the following formula:

$$RPP = (WKL1\% \times ITRR_{WKL1} + WKL2\% \times ITRR_{WKL2} + \cdots + WKLn\% \times ITRR_{WKLn})$$

where $WKL1\% + WKL2\% + \cdots + WKLn\% = 100\%$. CP2000 uses a similar but more refined metric called the M-value, which we will define shortly.

CP2000 Uses LSPR Data

The LSPR results are used in CP2000 (Capacity Planning 2000). Users of CP2000 can optionally specify their own workload to get the most reliable comparisons between current and future systems. The default workload mix used in CP2000 is composed of equal parts of CICS, TSO and IMS (DB/DC).

LSPR Documentation

For more information, refer to *Large Systems Performance Reference (LSPR)*, SC28-1187. Also refer to A.1, "Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)" on page 249.

More Information about Parallel Sysplex

There is no specific LSPR documentation related to Parallel Sysplex. However, information on performance in a Parallel Sysplex is found in *S/390 MVS Parallel Sysplex Performance*, SG24-4356 and *DB2 for MVS/ESA V4 Data Sharing: Performance Topics*, SG24-4611.

LSPR ITRR Expression for a Mixed Workload

LSPR recommends using a harmonic mean for mixed workload ITRRs. Compared to the simple weighted average used when you calculate RPP values, the harmonic mean method gives a higher accuracy, especially when the ITRR values differ significantly between the workloads. The expression for harmonic mean is shown in the following formula:

$$ITRR_{mixed\ workload} = \frac{1}{\left(\frac{WKL1\%}{ITRR_{WKL1}} + \frac{WKL2\%}{ITRR_{WKL2}} + \cdots + \frac{WKLn\%}{ITRR_{WKLn}}\right)}$$

Where

$$WKL1\% + WKL2\% + \cdots + WKLn\% = 100\%$$

While the harmonic mean may seem an obscure function, it is encountered in everyday life. For example, if a car travels 50 km at 50 km/h and a further 50 km at 100 km/h, then the average speed of the car is the harmonic mean of 50 and 100 km/h, (66.6 km/h), in other words not their arithmetic mean (75 km/h).

This example illustrates a fact about the harmonic mean of positive numbers: it is always less than or equal to the arithmetic mean.

Harmonic Mean: an Example

To consider an example closer to home, imagine two workloads on two processors. Suppose that we run the first processor with equal processor load from each workload. Suppose the second processor runs transactions from the first workload twice as fast as the first processor, and transactions from the second workload ten times as fast as the first processor. What is the relative processor power for these two processors for this workload mix? A reasonable approach would be to calculate how long the second processor would take to complete the transactions that the first processor would run in one hour, and divide the two times. The second processor takes 15 minutes to complete the first transaction type and 3 minutes to complete the second transaction type. The relative power is then 60/(3+15)=3.33, which is the harmonic mean of 2 and 10 and is very different from the arithmetic mean (which is 6).

As a final comment, suppose we were to ask what the relative power of the two processors is for a 50-50 workload split on the second processor. How long

would the first processor take to process 30 minutes worth of the first workload and 30 minutes of the second? Clearly 30x2+30x10=360 minutes. The relative capacity is now 6 - the arithmetic mean! In fact, the first processor processes the first workload at .5 the rate of the second processor, and it processes the second workload at .1 the rate of the second processor. The harmonic mean of .1 and .5 is 1/6, and hence the second processor has 6 times the power of the first. These relationships are no coincidence; similar results would hold with any other values for relative speed.

Chapter 7 of the book by A.O. Allen justifies the use of the harmonic mean more formally in Proposition 7.3.2.

CP2000 M-value

CP2000 uses an internal value for CEC power. This value is called the M-value. The M-value is a special case of the mixed workload ITRR formula.

The number of I/Os is scaled according to the M-value using the Relative I/O Content (RIOC). Refer to 1.4.1, “Relative I/O Content (RIOC)” on page 31 for a discussion on the relationship between I/O and the M-value. The M-value, shown in the following formula, is based on LSPR ITR numbers. For OS/390, equal parts of TSO, IMS/DS, and CICS are used. A harmonic mean for the workload ITRRs is derived and then multiplied by a constant to get the M-value. This is illustrated in the following formula:

Relationship between ITR and M-value

$$M\text{-}value = \frac{3}{\left(\frac{1}{ITRR_{CICS}} + \frac{1}{ITRR_{TSO}} + \frac{1}{ITRR_{IMS}}\right)} \times F$$

The scaling factor F has a value of 744.35, which is the M-value of an IBM 3090-180E.

An M-value calculation example for a 9672-R52 CEC follows:

$$M_{9672\text{-}R52} = \frac{3}{\left(\frac{1}{5.39} + \frac{1}{5.23} + \frac{1}{4.55}\right)} \times M_{3090\text{-}180E} \Rightarrow$$

$$M_{9672\text{-}R52} \approx 3744$$

Note: The ITRR values are from LSPR and can be obtained via the tools PCR or LSPR/PC, which are discussed in A.1, “Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)” on page 249.

As mentioned before, the M-value is based on equal TSO, CICS, and IMS/DS workloads.

M-value and Other Workload Composition

A calculation carried out for the same CEC, but using batch (CBW2), TSO, CICS, IMS/DS, and DB2 prorated, shows a difference of less than 2% for this particular case. This is not significant. You should simply be aware of this effect and make the necessary changes as needed.

In case you rerun a model created by prior releases of CP2000, the M-value is recalculated and adjusted based on up-to-date LSPR values. CP2000 provides the user with the facility to define and store M-values.

How to Obtain M-values for Different CECs

How to obtain the current M-values are discussed in Appendix D, “M-value Table (December 1997)” on page 293.

The number of I/Os is scaled according to the M-value using the Relative I/O Content (RIOC). Refer to 1.4.1, “Relative I/O Content (RIOC)” on page 31 for a discussion on the relationship between I/O and M-value.

1.3.2 OS/390 CPU Time Account and Capture Ratio

OS/390 accounts for the following CPU time:

- Task Control Block (TCB) time on an address space basis
- Service Request Block (SRB) time on an address space basis
- Region control task (RCT) time on an address space basis (Goal Mode only)
- I/O interrupt (IIT) time on an address space basis (Goal Mode only)
- Hiperspace service (HST) time on an address space basis (Goal Mode only)
- CPU wait time per CP

Non-Captured Time

If you sum all the accounted address space time, you get something less than the total CPU busy time. The difference is called *non-captured* time.

Captured Time

Captured time includes CPU time consumed by services such as JES, Allocation, Master Scheduler, VTAM, RMF, and so on. Some captured time is considered as non-captured because it is not shown on an address space basis in the RMF Workload Activity report (SMF type 72), but it is charged to the address space (in SMF type 30). Examples (in Compatibility Mode) include:

- Region control task (RCT), mainly for swap activities
- I/O Second Level Interrupt Handler (SLIH)
- Non-MVPG hiperspace movement
- CPU time to find a specific frame to hold a page fix request (reported in SMF type 71)

Some OS/390 code is not CPU-accounted and therefore not charged to anyone. This time belongs to *non-captured* time and includes:

- Overheads such as page faults and some swapping activities
- Global functions that cannot be charged to a user, such as the OS/390 Dispatcher or some SRM logic
- I/O First Level Interrupt Handler (FLIH), because there is no knowledge of who the I/O requester is at this point in time

In the UNIX world, the CR is not a topic of big debate. In a UNIX system there is usually only one main application in the CEC. Then the total CPU power used by the application is the total CPU power used in the CEC.

How To Prorate Non-Captured Time?

Non-captured time is derived by subtracting captured time from CPU busy time, as discussed later. An important capacity planning question is how to prorate non-captured CPU time and service CPU time among workload components. As you will see, this is not really a problem.

Parallel Sysplex Non-Captured Time

Measurements from Parallel Sysplex data-sharing environments show that the percentage of non-captured time *decreases* slightly (1-2%) compared to non-sysplex environments. This is because almost all Parallel Sysplex cost, including data-sharing cost, is captured. Parallel Sysplex captured and non-captured times are discussed in more detail in the following section.

1.3.2.1 Capture Ratio

Capture Ratio Definition

Capture Ratio (CR) is the ratio of the captured time to the total CPU busy time.

$$Capture\ Ratio = \frac{Captured\ Time}{Total\ CPU\ Busy\ Time}$$

The concept of capture ratio also applies to I/O and storage. If you have a low capture ratio, you need to do some further calculations based on certain assumptions. We will discuss some of these assumptions in following sections.

One way to describe the concept of capture ratio is to look at a simple example. The following example only contains two applications to make calculations easy.

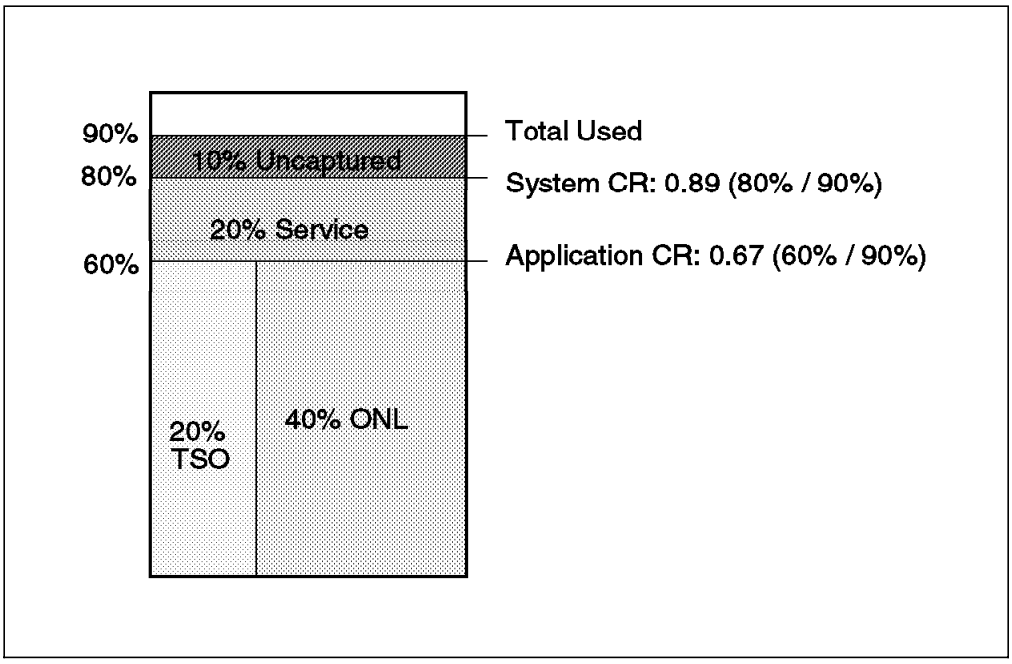


Figure 3. Captured and Non-Captured CPU Utilization

These Numbers Appear in RMF Reports

Figure 3 shows two workloads, TSO and ONL, in a system. There are no other applications on this system. The total CPU load is 90%. There are service functions using 20% CPU, related to services done by OS/390, JES2, VTAM and so on. There is some CPU usage (10%), but it cannot be specifically associated with any workload (non-captured).

The calculations for the capture ratios are as follows:

System Capture Ratio

$$\text{System capture ratio} = \frac{80\%}{90\%} \approx 0.89$$

The system capture ratio tells how much of the total CPU usage is captured. For capacity planning purposes, this capture ratio *is less important.* It cannot be used for recalculating the total CPU usage of an application.

Note: The system capture ratio is often used for capacity planning. However, the application capture ratio is a better alternative.

Application Capture Ratio

$$\text{Application capture ratio} = \frac{60\%}{90\%} \approx 0.67$$

The application capture ratio tells how much of the CPU usage is captured to the applications in the system, in relation to the *total CPU load.* For capacity planning purposes, this capture ratio is *very valuable.* It is used to recalculate the total CPU usage of an application.

The application's CPU consumption is derived using the capture ratio as follows:

- $\text{TSO total CPU usage} = \frac{\textit{TSO Usage captured\%}}{\textit{Application Capture Ratio}} = \frac{20\%}{0.67} \approx 30\%$
- $\text{ONL total CPU usage} = \frac{\textit{ONL Usage Captured\%}}{\textit{Application Capture Ratio}} = \frac{40\%}{0.67} \approx 60\%$

Note that the non-captured time and captured service address space CPU time is prorated *evenly* among the applications. This might not always be the best approach. The following are some different methods to prorate non-captured time:

1. Apply the non-captured time uniformly to all workload elements in relation to their CPU utilization. This is a very simple solution and works fine when the capture ratio is high, or the workload elements are similar in nature. *This is the default method used by CP2000.*
2. Sometimes non-captured time is divided in relation to the number of I/Os issued by the application. This is often a recommended solution and is a fairly simple and accurate method. CP2000 allows you to use this method.
3. Prorate non-captured time by I/O rate *and* account for the following services as described here:
 - System services to *all* workloads
 - VTAM CPU time to TSO, CICS and IMS
 - JES CPU time by TSO and batch

 This is a solution that is sometimes recommended in systems where very different workloads exist. You can use this method with CP2000 if you do some manual calculations yourself.
4. Provided you have several measurements from the system and assuming the capture ratios are constant for the workloads, sometimes it is possible to do division by using mathematical formulas (linear regression techniques). This is undoubtedly the most complex algorithm and is not generally recommended. CP2000 allows you to use this method.
5. You can also apply your own capture ratio to the workloads based on feeling or experience. This is a simple approach, but is only recommended if you know what you are doing. CP2000 allows you to use this method.

Only in cases with *low or very different* capture ratios should you consider methods 3 through 5. Figure 4 on page 17 shows the applications, TSO and ONL, and their total CPU usage after capture ratios were applied. For simplicity, we distributed the non-captured CPU time evenly (method 1).

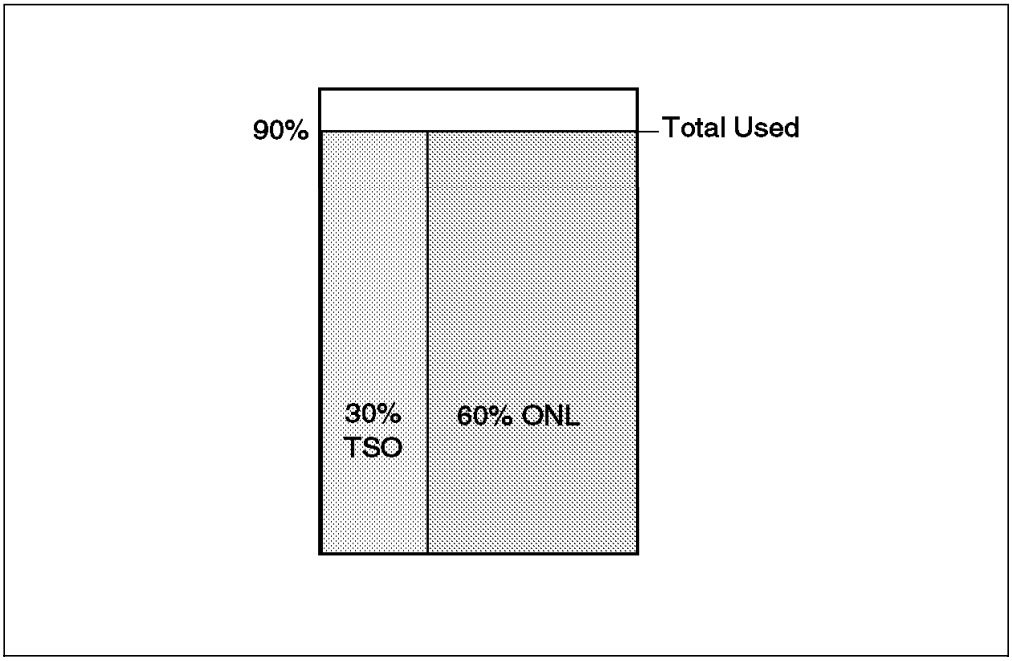


Figure 4. CPU Utilization after Applying CR

LSPR and Capture Ratios

In the LSPR methodology, since each benchmark is of a single workload, the total CPU utilization is attributed to that workload. In other words, the LSPR benchmarks do not directly concern themselves with the capture ratio or captured service CPU time. When we use the harmonic mean formula to estimate the capacity for a mixed workload, we get built-in estimates for uncaptured CPU time and service address space CPU time appropriate for each workload in that combination.

LPAR Management Time

This example does not cover LPAR. When using LPAR, the LPAR Management Time must also be prorated to the applications.

Causes of High Percentage of Non-Captured Time

There are many reasons why the *non-captured time may increase.*

The following list shows common reasons for high or very high percentages of non-captured CPU time:

- **High I/O load**

 I/O interrupts are processed by the First Level Interrupt Handler (FLIH) routine. This time is not going to be accounted to the I/O requestor. At this stage the system does not know to whom the I/O belongs.

- **High paging rate**

When paging occurs, the associated CPU utilization is not charged to anyone. Most of the CPU time is charged and accounted as SRB time in Performance Group Number 0 (PGN 0).

 - **High page movement within CS**

 Some of the CPU time used to move pages *within* CS storage is not captured to any TCB/SRB.
 - **High CS-ES page transfer rate (lack of CS)**

 Some type of Central/Expanded Storage page transfer rate, such as page stealing, is not charged to anyone, as the system is serving all the workloads.
 - **High CS UIC update rate (lack of CS)**

 This time is not charged to anyone, as the system is serving all the workloads.

- **Spin lock recovery**

 When spin lock processing occurs, the associated CPU utilization is not counted to anyone.

- **TSO logon processing**

 Part of the TSO logon processing is not accounted to anyone, partly because it has not yet been established who the user is.

- **Many TCBs per address space**

 Specialized applications having many TCBs in the address space may bring down the capture ratio due to dispatcher activity. This effect is much reduced in MVS V5 and later releases due to the dispatcher changes. For example in the address spaces of OS/390 UNIX Services applications there are typically several TCBs due to usage of threads (3.5.2, “Threads” on page 111).

- **Traces**

 Active traces and traps, especially Program Event recording (PER) type Serviceability Level Indicator Processing (SLIP) traps, can cause very low capture ratios.

- **Operating system errors**

 Some times errors in operating system code have been known to lower the system capture ratio.

- **LPAR management**

 LPAR management time can increase the non-captured time especially in systems prior to the LPAR enhancements in MVS V4.3. Low utilization effects in LPAR environments running shared CPU configurations can cause low capture ratios.

Causes of High Service Time CPU Utilization

There are many possible reasons the captured CPU time for the *service functions may increase.*

Below is a list that identify common causes of high or very high percentages of captured CPU time for service functions:

- **High monitoring activity**

If there are many monitoring programs that consume CPU resources, fewer resources are available for the remaining workload. When you monitor CPU utilization at a very detailed level, it may lead to a heavy CPU load.

- **Many batch or OLTP programs**

 There are many batch or OLTP programs, and each controlling service program (VTAM, JES2 and so on) also uses the CPU.

- **Many small steps in batch jobs**

 There are many small steps in your batch jobs, which in turn leads to high allocation costs.

- **Badly tuned system**

 The system is not properly tuned or set up. SRM parameters, LPAR definitions and so on can affect the effectiveness of the system and also affect the capture ratios.

- **Data In Memory**

 Data in Memory techniques, such as Batchpipes/MVS, reduce non-captured time since less I/Os are done, thus increasing capture ratios. This gives better accuracy to application accounting of CPU.

Typical Capture Ratio Values

Typical values for the capture ratios in an OS/390 environment:

- For the system capture ratio, typical values are around 0.85 - 0.92.
- For the application capture ratio, typical values are around 0.65 - 0.80.

Typical application capture ratio values depend on many factors. Table 1 gives you some guidance about typical application capture ratios.

Table 1. Typical Application Capture Ratios for Different Workloads (1997)

Workload type	**Capture Ratio**	**Remark**
Short-running batch	0.75 - 0.85	
Long-running batch	0.80 - 0.90	
Engineering/Scientific batch	0.85 - 0.95	
Light CICS transactions	0.75 - 0.85	lower in the DB2, MRO environment
Heavy CICS transactions	0.80 - 0.90	lower in the DB2, MRO environment
Light IMS transactions	0.70 - 0.90	lower in the DB2, cross-memory environment
Heavy IMS transactions	0.75 - 0.95	lower in the DB2, cross-memory environment
Light TSO	0.60 - 0.70	
Normal TSO	0.65 - 0.75	
Heavy TSO	0.75 - 0.85	

Note: Typical transaction path lengths are:

- DB2 1.000.000 to 3.000.000
- IMS 750.000 to 2.000.000

In a Parallel Sysplex, the method for calculating the capture ratio will not change. In general all services related to Parallel Sysplex (primarily XCF and

XES type services) are captured. Depending on the subsystem and workload characteristics, it is either the requestor or the system that this time is accounted to.

Parallel Sysplex Capture Ratio Values

Parallel Sysplex Capture Ratios

Measurements indicate that the *system capture ratios* for systems participating in Parallel Sysplex data sharing *increase 1-2%.*

In systems where most of the CF activity is synchronous and there is little coupling link contention, then the *application capture ratios* tends to *increase 1-2%.*

In less tuned Parallel Sysplex configurations experiencing many asynchronous CF accesses and/or many subchannel busy conditions, the application capture ratio *drops slightly.*

In case of many asynchronous CF accesses it is possible to prorate the captured system CPU time to applications or subsystems in relation to their CF request rates.

1.3.3 Balanced System Concept

To do capacity planning, you need sample data from an *unconstrained* system; it is not desirable to conduct a capacity planning study based on data coming from a constrained system. Constrained systems may present latent demand that cannot be satisfied. At best, using historical analysis, we can perhaps have some insight into the latent demand.

Unconstrained Systems Are Key for Reliable Capacity Planning

A constrained resource in a system can mask a constraint for another resource. To show this effect, consider two examples:

- A system is delayed for DASD reasons. You cannot uncover what happens if the DASD pain is relieved. Will the DASD bottleneck be replaced immediately by a CPU bottleneck or by a storage bottleneck, or will the system begin to page?
- A dedicated TSO system running at 90% CPU utilization with 1000 TSO users is seen by capacity planning tools as a system that needs a little less then 0.090% CPU utilization for each TSO user.

 If this system is analyzed at a time when 2000 TSO users are logged on, the system will probably show 100% CPU busy. The capacity planning tool will conclude that each TSO user (experiencing a worse response time) needs:

 $\frac{100\%}{2000} \approx 0.050\%$, which differs significantly from 0.090%.

Projections based on the two examples will give different results, even though the projection covers the same CEC and the same kind of TSO users.

Note that the two examples mentioned above can lead to the general conclusion that all resources should be contention free. This is of course not always possible, so maybe a better approach is to state that no resource should experience severe delays.

Latent Demand

Latent demand is the workload that did not run because of missing resources. The constraint may be one of policy. For example, a certain type of workload cannot be processed on the first shift, or the constraint may be due to poor response time or to a lack of an adequate number of terminals. It is *impossible* to measure the latent demand, but it is possible to verify its non-existence.

One consequence of latent demand is the leveling of the load. The peaks cover the valleys. We take a closer look at this effect in 1.3.4, “Peak-to-Average Ratio (PAR) and Saturation Design Point (SDP).”

One aspect of latent demand you will not be able to estimate is the human factor. When some resource (for example the CPU) gets so highly used that it affects response time (or turnaround time), people are not able to do as much work as they would like to accomplish. They enter fewer transactions or batch jobs or do not create new work. Your business might be hurt.

This effect causes a *pent-up* demand that becomes a real demand on the system if or when the bottleneck is cleared up and response time (turnaround time) gets better. You can only estimate the extent of the pent-up demand from the duration of the existence of the bottleneck. If the bottleneck has been around for only a few days and then disappears again, there is little pent-up demand. If the bottleneck has been around for a year, be careful. In a Parallel Sysplex exploiting dynamic workload balancing, a peak on one system, can to an extent, cover a valley on *another* system. To a certain extent this is even true for “static” workload balancing. This means that in the Parallel Sysplex the possibility exists for a better usage of the available CPU resources.

Latent Demand and Parallel Sysplex

Parallel Sysplex can, to a higher degree, eliminate the effects of latent demand due to dynamic workload distribution capabilities. This is true if workload balancing is supported and implemented for the transaction manager. The latent demand phenomena still exists and should be avoided. If latent demand occurs due to a resource constraint on a system, the workload can be routed to a non-constrained system in a Parallel Sysplex environment.

It is always useful to run a reality-check analysis before a capacity planning study. A reality-check may:

- Reveal presence of latent demand.
- Identify major bottlenecks that are of a temporary nature.
- Identify loops and the like that will tend to use the remaining capacity of a CEC. If this is not part of the dimensioning workload, it should either be factored out or the measurements should be repeated.

Such a check is for example done with Capacity Planning Automated Input Processor CPAIP, formerly known as ARIAS.

1.3.4 Peak-to-Average Ratio (PAR) and Saturation Design Point (SDP)

The *Peak-to-Average Ratio (PAR)* is the ratio of *peak* CPU utilization to *average* utilization. Usually the peak CPU utilization represents a one hour period, and the average utilization an eight hour prime shift. Pay attention that peak CPU is the *highest* average of the set of one hour periods sampled. You may have reasons to use time periods other than one and eight hours. They tend, however, to be suitable for many installations.

PAR Formula

PAR is calculated as:

$$PAR = \frac{Peak_{CPU\ Utilization}}{Average_{CPU\ Utilization}}$$

PAR sometimes is also called “skew.”

SDP Formula

The *Saturation Design Point (SDP)* is an installation defined CPU utilization level, expressed as a percentage of the capacity of the system, and is calculated as:

$$SDP = \frac{100\%}{PAR}$$

Another way of stating this is to simply say that when your total CPU average utilization reaches the SDP, *the peaks will be at 100% CPU load.* If the system runs over the SDP on average, the system will probably start to experience latent demand.

The formula for SDP uses 100% CPU utilization as the upper limit for the SDP*PAR calculations. The assumption is that CPU is always the first bottleneck reached as the system load increases, and that CPU load can reach the 100% level. This is often true, but not always the case. End-user response time may begin to decline much earlier if there is a bottleneck somewhere else in the system. For example the bottleneck may be in the I/O subsystem or in the storage area. If there are no ″optional″ or ″non-critical″ workloads, then the 100% CPU limit cannot be reached without suffering response time degradation. As a bottom line, instead of 100% CPU utilization, use that CPU utilization at which end-user response time exceeds an acceptable level as your SDP.

Dimensioning Workload

The SDP is a value used to define your saturation limit associated with your *dimensioning workload*. If your workload (or the dimensioning part of it) on the average exceeds this limit, you might experience unacceptable response time.

There are no rules of thumb that apply to all systems. SDP depends on the variation of the workload over time and your experience based on response time. CP2000 uses a default value for SDP of 70%.

Note: The discussion of SDP presented here does not take into consideration such items as backup capacity. If this is an issue, your capacity planning should cater to that.

It is important that the peak measurements of your workload *are based on a reasonable time period* (for example an hour). If your peak measurements are averaged over shorter time periods, your SDP should be adjusted accordingly (upwards). If your peak measurements are averaged over longer periods, your SDP should be adjusted accordingly (downwards).

If you focus on response time, we recommend you use the CPU utilization threshold and then use an *Erlang Response Time Graph* to choose a suitable SDP. To see examples, refer to Figure 8 on page 26 and Figure 9 on page 28. Several reasons you might change the default SDP are outlined later in this section.

A very low PAR (close to 1), where the average CPU utilization tends to be very close to the peak CPU utilization, usually indicates a system with latent demand.

Historical peak-to-average ratios may be used to get a view of the latent demand. A unconstrained PAR is usually between 1.4 to 1.7 for the hours of an average day of a peak period (for example a week). A PAR of 1.4 corresponds to an average of 70% for peak of 100%, which is the CP2000 default.

To show some of the aspects being discussed here, look at Figure 5, Figure 6 on page 24, and Figure 7 on page 25. Now let us analyze what happens to an installation workload profile over time. Three scenarios are discussed:

- A *lightly loaded* system
- A *saturated* system
- An *overloaded* system

Let us see what happens to the SDP in the three scenarios.

Average CPU Utilization Below SDP

No Latent Demand

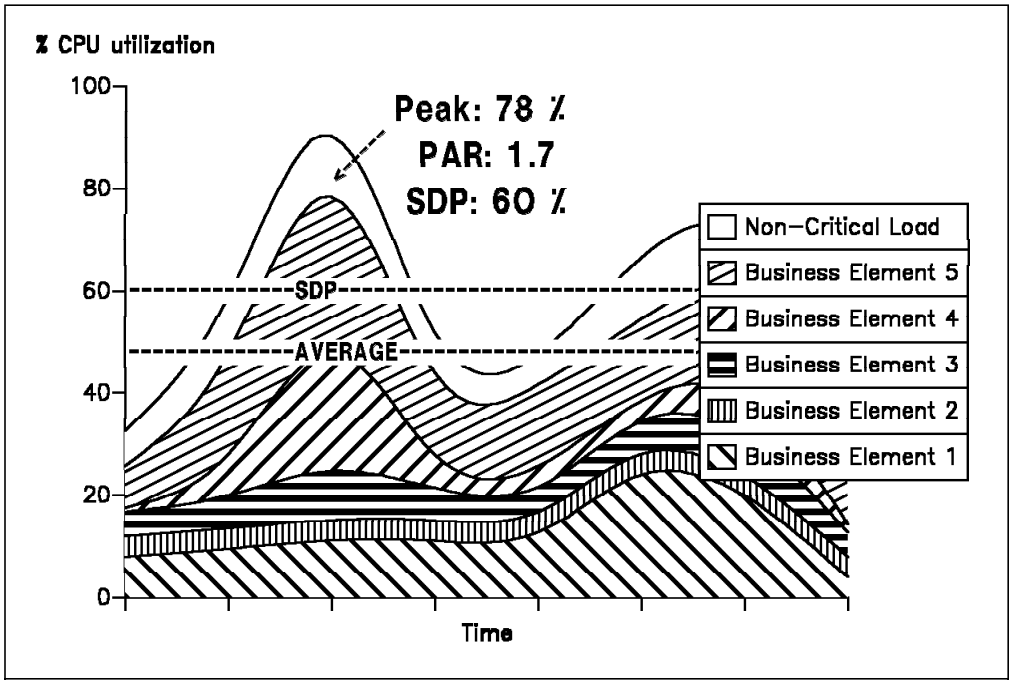


Figure 5. CP Load Distribution - Light Load

Figure 5 shows the workload distribution for a well defined period, for example, the prime shift. The graph shows a great variance in the total workload over time. Let us assume that we have used some capture ratio technique to derive the workload CPU utilization. The measured workload is composed of several components, of which most of them (Business elements 1 to 5 in this example) together comprise the dimensioning workload.

Figure 5 also shows the non-critical workload. The CPU consumption of this workload element is ignored in the evaluation, since it is not part of the dimensioning workload. Often non-critical workload is run at a low dispatching priority (DP) or in a discretionary service class. This means that the workload only runs when there is sufficient capacity to service the dimensioning workload. The non-critical workload represents workload that might not be performance sensitive, or is of less business value to the company. An example is low priority batch.

The saturation design point for this particular workload is set to 60% and its related current peak is measured at 78% CPU utilization. The peak is the

average CPU utilization over the peak period. The average CPU utilization in this example is below the SDP and is measured to 50%. We can see that all the current workload, whether critical or not, is served by the system, and there seems to be neither latent demand or CPU capacity constraints.

Now, move on to the next figure.

Average CPU Utilization Equals SDP

No Latent Demand?

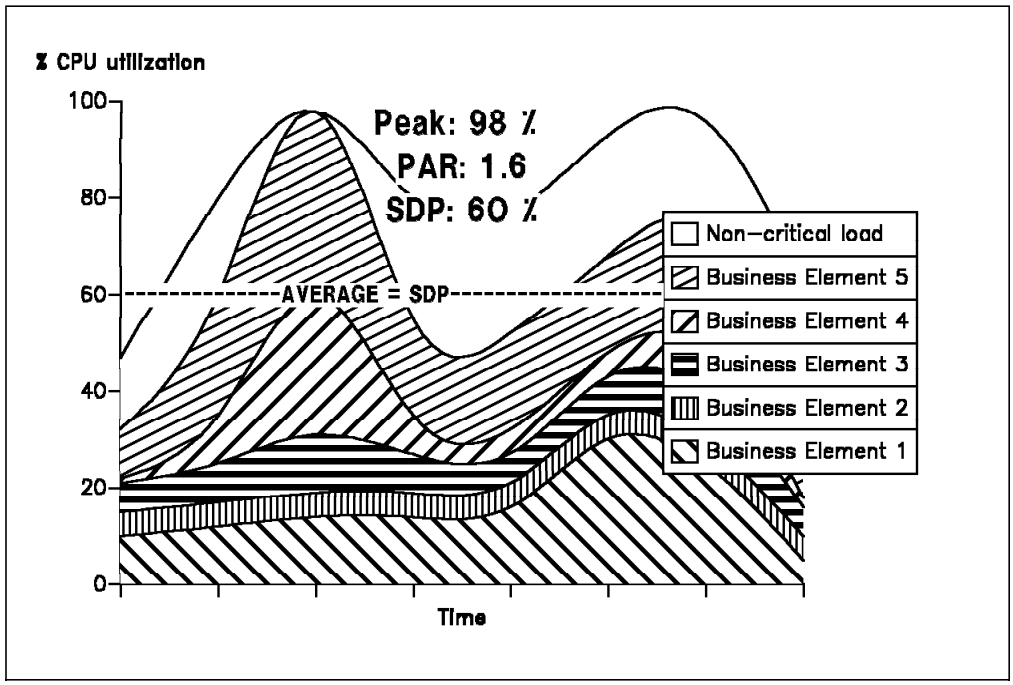


Figure 6. CP Load Distribution - Maximum Load

In Figure 6 we can see what happens to the workload when the average CPU utilization has increased to an average equal to 60% (equal to SDP).

The example shows a peak of 98% CPU utilization. Note that the non-critical load is “squeezed” away from the peaks. This system may need further resources for growth. If this is not granted, the situation is depicted in the next graph.

Average CPU Utilization Above SDP

Latent Demand

Levelling of Workload

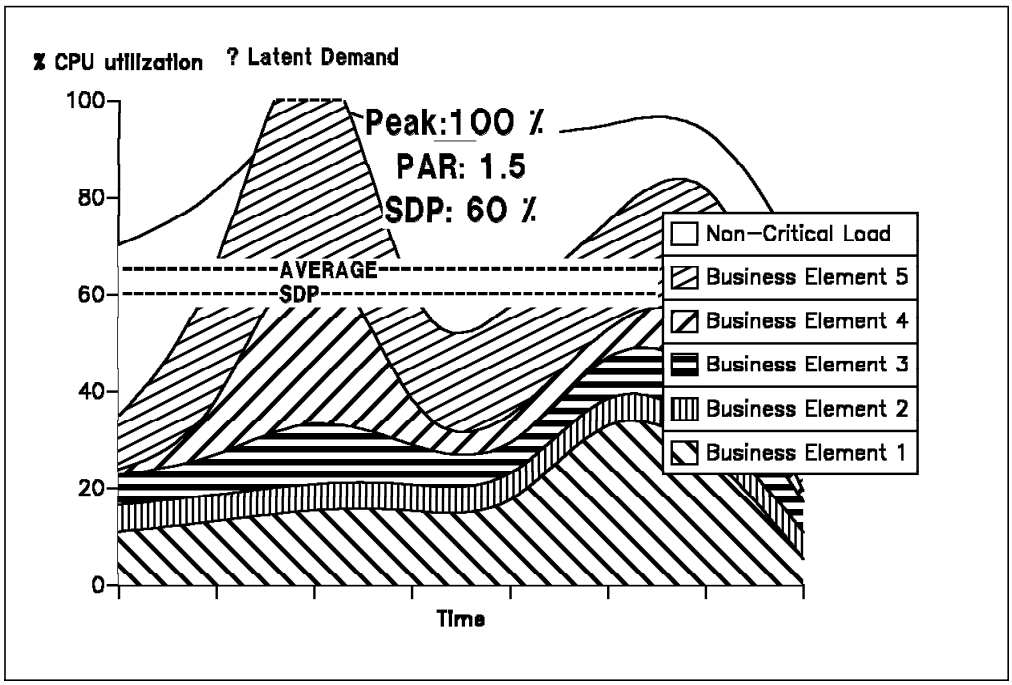


Figure 7. CP Load Distribution - Overload

Figure 7 shows that the average CPU utilization is now above the SDP and is 65% *for the dimensioning workloads.*

This system experiences latent demand. Observe that the total CPU utilization reaches close to 100% over longer periods of time.

1.3.4.1 Measuring Peaks Only

Many installations base their capacity plan on peak periods only, and do not consider the PAR (peak-to-average ratio). The peak periods must be carefully selected. One possibility is to consider the peak of the hourly weekly averages.

1.3.4.2 Workload Collision Analysis

When workloads from several system are combined, the PAR of the combined system tends to be less than the PAR of the component systems because the peaks in the different workloads on the different systems do not always coincide. In Parallel Sysplex the workloads are handled as one entity, balanced between the participating systems and the peaks are lower ones (PAR ratio is lower).

There is no mathematical formula to calculate the smoothed PAR. But CP2000 provides a method to see all the workloads together in the Parallel Sysplex. This method is called *collision analysis*.

1.3.4.3 SDP Considerations

In the three examples just shown, we did not explain how, and for what reason, we chose to set the SDP at 60%. You choose a particular SDP to eliminate or diminish the effects of queuing. There are different methods to determine your SDP:

- The default SDP taken by CP2000 (70%)
- The user chosen SDP is based on data related to the installation such as:
 - Variation in CPU around average
 - Response time factors

Your customized SDP can act as a response time threshold.

We need a threshold to decide when to change or upgrade the CEC because as a CEC becomes more busy, the application response time grows because of an increase in the queue time as illustrated in Figure 8. The SDP is governed by several factors:

- Your own experience
- The number of central processors (CPs) in the CEC
- The CPU power per central processor (CP) in the CEC
- The total CPU power in the system
- Variation in CPU around average
- The degree of dynamic workload distribution capability in a Parallel Sysplex

Without getting into too much detail, we can also deduce SDP by looking at the following charts:

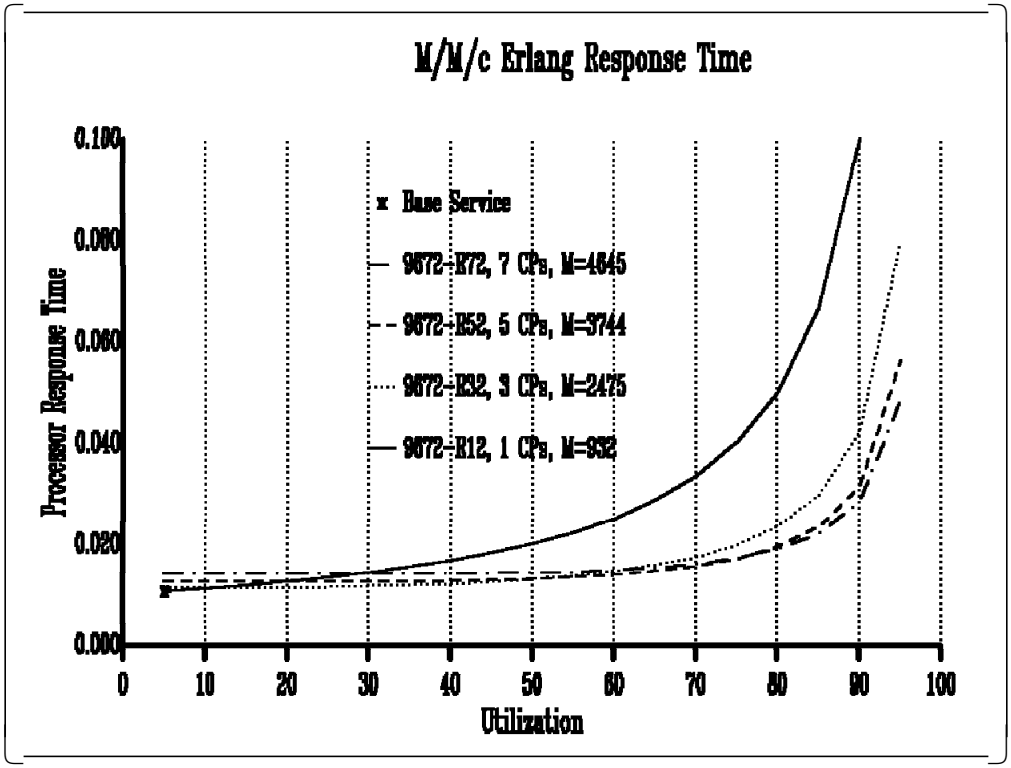


Figure 8. CP Response Time versus Utilization for Various Number of CPs

Figure 8 shows on the Y axis the CP response time given the CP service time is equal to approximately 0.01 seconds. This and other similar graphs are seen in CP2000, option 7.1.4.

The CP response time, T_r, is based on the following formula:

$T_r = T_q + T_s$ where

T_q is the CP queueing time and
T_s is the CP service time

In the chart the M-value for the illustrated CECs are also shown. The X axis shows the total CPU utilization expressed as percentages. The heading M/M/c refers to the statistical method behind this set of curves and is often referred to

as “Markow theory.” The first M refers to the “exponential interarrival time distribution of transactions,”

The second M refers to the “exponential distribution of service times” and *c* stands for the number of servers (or CPs in our case).

Complex systems are modelled according to these statistical methods with a certain degree of confidence. M/M/c response time distribution imply that all incoming transactions have the same priority in the system. Since OS/390 implements a *priority scheme,* high priority jobs will not increase their response time as rapidly as Figure 8 on page 26 indicates. For further discussion on this, refer to:

- 7.2, “Case Study: Analyze Processor Speed” on page 215
- 7.2.3, “Assumptions and Restrictions for the M/M/c Queuing Model” on page 223
- Chapter 5, “A CF Processing Model” on page 143
- Appendix E, “Queueing Theory Concepts” on page 297

The graph shows that as the number of CPs increase, it is possible to increase the total CPU utilization *with less* queuing penalty.

Now let us try to be precise by looking at some examples:

- Choose a SDP of, let′s say, 90%. Figure 8 on page 26, tells you to expect the CPU response time to be close to:
 - 0.10 s for one CP
 - 0.09 s queue time
 - 0.01 s service time
 - 0.04 s for three CPs
 - 0.03 s queue time
 - 0.01 s service time
 - 0.03 s for five CPs
 - 0.02 s queue time
 - 0.01 s service time

- Let us look at it from a different perspective. Let′s pretend you accept a CPU response time of, let′s say 0.02 seconds. The SDP is now around:
 - 45% for one CP
 - 75% for three CPs
 - 80% for five CPs

Observe that the CPU Service Time is slightly different among those CECs and it is lowest with one CP.

SDP is a Function Of The Number Of Servers

So the conclusion is to let the number of CPs in your system be one of the factors you consider when making your choice of SDP.

The previous figure may give a misleading picture when the comparison is made for CECs that do not have the same kind of engines. What happens if we look at processor complexes having almost identical CEC power, but with a *different number of CPs?* This is illustrated in Figure 9 on page 28.

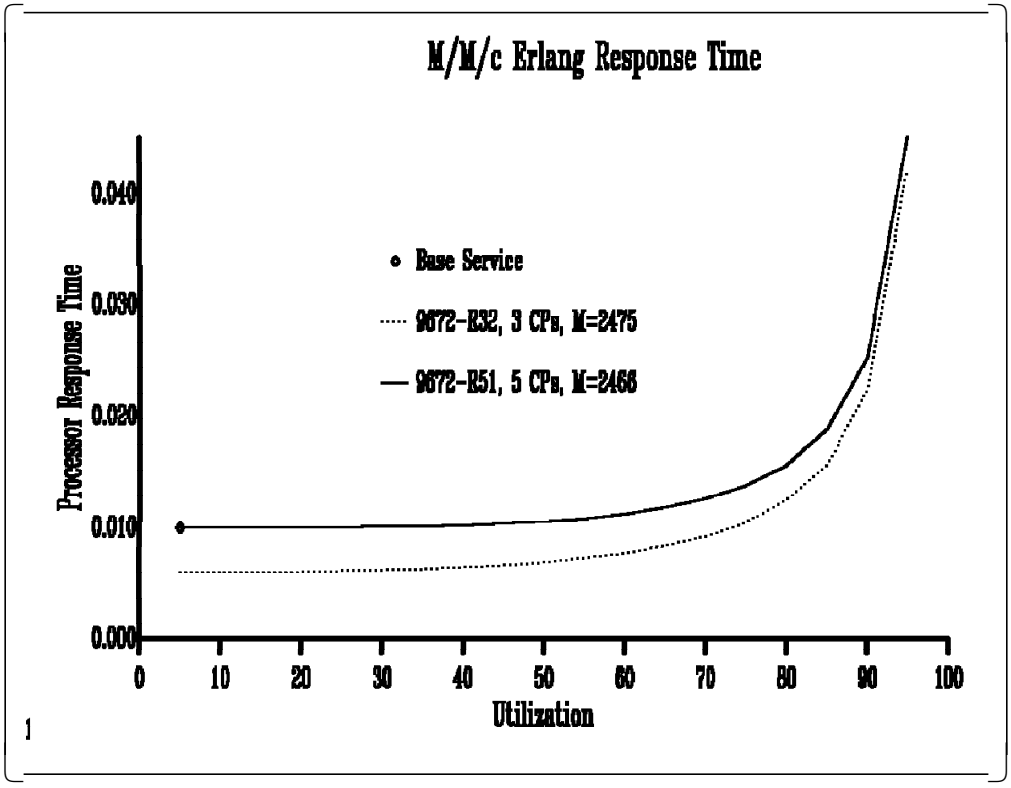


Figure 9. Response Time over Utilization As an Effect of CP Speed

Figure 9 shows, using the same statistical method as before, the CPU response time for two systems. This graph is seen in CP2000 through option 7.1.4.

The two CECs here have about the same total power, in this case expressed as having similar M-values. One CEC has three CPs and the other has five CPs.

The Y axis shows the CPU response time. The X axis shows the total CPU utilization expressed as percentages.

The conclusion is that the system with fewer and faster CPs (given the same total power) is at least equally responsive to the system with more and slower CPs. The total response time for a job or transaction is only partly influenced by the CP response time, so often this is of less importance. For some applications whose SDP target are high (70% - 90%) the MP solution might provide better performance than an UP even if the single MP CP is less powerful than the UP engine. In cases where you need a deeper understanding of this effect on, for example, your batch jobs, refer to tools that can help you. Batch Workload Analysis Tool (BWAT) and other tools that help to evaluate effects of migration to CECs with smaller engines are listed in A.6, “Enterprise Server Capacity Planning Tools Catalogue” on page 271.

Note that with the help of CP2000, you can model these effects for the SDP value. It is possible for you, as a CP2000 user, to override the default SDP value of 70%. However, this new value will remain constant for all the projected scenarios, no matter how many CPs are in the projected system. The concept for SDP and PAR are only slightly affected by the Parallel Sysplex. What is going to change is the system behavior due to the capability of balancing the workload across the entire complex as soon as a constraint in CPU resource occurs.

1.3.5 MP Factor

The Multiprocessing (MP) tightly coupled architecture does not have good scalability. This means that when you add more CPs sharing the same memory, the gain in performance is not going to be linear with the amount of added hardware. Figure 11 on page 47 depicts the MP effects when adding additional CPs in an MP environment.

This lack of scalability is sometimes referred to as the MP factor. The MP factor is a function of:

- CPU caching (the likeliness of High Speed Buffer (HSB) misses increases with additional CPs)
- Access to storage (can be affected by the number of CPs)
- OS/390 locking
- Signalling (between CPs increase non-linearly when more CPs are added)

A good hardware design can decrease the MP factor, such as in 711-based CECs. However there are limits, primarily related to cost, to how far this design can be improved.

In Chapter 2, “Capacity Planning in Parallel Sysplex” on page 45 we discuss how scalability improves dramatically in a Parallel Sysplex.

1.3.6 SRM CPU Service Units

Systems Resources Manager (SRM) uses the service definition coefficients to assign additional weight to one type of service relative to another, allowing the installation to specify which type of resource consumption should be emphasized in the calculation of service rates. For example, an IPS may contain the following service definition coefficients:

```
CPU=10.0    IOC=5.0    MSO=0.0    SRB=10.0
```

Let us take a quick look at the different categories of service units, for more information refer to *OS/390 MVS Initialization and Tuning Guide*, SC28-1751.

- **CPU service units** relate to task (TCB) execution time, multiplied by an SRM constant which is CPU model dependent. Included in the execution time is the time used by the address space while executing in cross memory mode (that is, during either secondary address mode or a cross memory call). This execution time is not counted for the address space that the target of the cross memory reference.
- **SRB service units** relate to service request block (SRB) execution time for both local and global SRBs, multiplied by an SRB constant which is CPU model dependent. Include in the execution time is the time used by the address space while executing in cross memory mode (that is, during either secondary addressing mode or a cross memory call). This execution time is not counted for the address space that the target of the cross memory reference.
- **I/O service units**. Refer to 1.4.2, “SRM I/O Service Units” on page 33.
- **Storage service units**. Refer to 1.5.4, “SRM Storage Service Units” on page 38.

If you do not define the service coefficients, the defaults are:

```
CPU=10.0    IOC=5.0    MSO=0.0    SRB=10.0
```

Since changing the coefficients affects the durations and the accounting values, the defaults are meant to be consistent with settings seen today.

However, when you plan to use the OS/390 Workload Manager (WLM) we recommend specifying the coefficient as follows:

```
CPU=1.0    IOC=0.5    MSO=0.0    SRB=1.0
```

By specifying the CPU and SRB coefficients as ones, there is consistency between the WLM resource group management values and RMF report values.

1.3.7 LPAR Capacity Planning

Parallel Sysplex often introduces an additional number of Logical Partitions (LPs). Some of these are for OS/390 systems and others may be for CFs running either ICMF or CFCC.

LPAR efficiency varies in addition to your setup also by CEC family. This is because of hardware design factors and microcode levels.

To provide a reasonable assessment of LPAR capacity, IBM developed the LPAR Capacity Estimator (LPAR/CE). LPAR/CE is available to your IBM representative. It is a PC based productivity tool for IBM CECs with the PR/SM feature. If there is a definite need to do LPAR capacity planning, then LPAR/CE is the best tool for that purpose. For additional information about LPAR/CE, refer to A.2, “LPAR/Capacity Estimator (LPAR/CE)” on page 253.

1.3.7.1 LPAR Performance Considerations

The effective LPAR capacity depends on many variables. To minimize overhead, consider the following general rules of thumb:

LPAR overhead is associated with the total number of LPs, dedicated versus shared CPs, ratio of logical to physical CPs and the type of SCP, workload, and environment.

- For 9672 and ES/9000 LPs, CPs are dedicated or shared.

 When a sharing logical partition is activated, it should be assigned enough processors to *meet its peak demands* and any immediate growth requirements. Too few processors could limit the number of potential transactions, and too many active logical processors could affect performance.

- The *Processing weights* are used to specify the portion of the *shared processor*. A logical partition with dedicated processors is not affected by processing weights.

- Usually, dedicated partitions have less overhead, while shared partitions offer the most flexibility.

- For a production CF LP, CPs should be dedicated.

 Because a CF LP uses all CP resources made available to it, and coupling links have critical response time requirements, the general recommendation is to dedicate CF CPs. However ICMF assist allows sharing of CPs between OS/390 (MVS) and a CF with a reasonable performance.

- The workloads best suited for logical partitions (using shared CPs) are those that have a wide fluctuation of external throughput or those that do not fit well into the capacity of a dedicated logical partition.

- Within a dedicated partition, do not assign more CPs than necessary.

 The number of processors required to meet the peak demand should be determined in the same manner as in basic mode.

- There is a direct relationship between the number of logical CPs and the LPAR overhead. More logical CP management increases overhead.
- You should have no capped logical partitions, unless you have planned requirements.
- The SRM threshold to control the number of CPs enabled for I/O interrupts CPENABLE should be set as follows:
 - ES/3090 and ES/9000, in LPAR-mode with shared CPs CPENABLE(0,0)
 - In all other case CPENABLE(10,30)

 The default settings are slightly different.

Note: Be aware that the default settings, especially for 9672, are not always optimal. 9672s in LPAR mode and using default settings, has been measured to have up to 10%+ ITR degradation for certain workloads. Sharing of resources is EC level dependent. See the latest version of *S/390 9672/9674/2003 PR/SM Planning Guide*, GA22-7236. for details.

Note: CF LPs using ICMF show increased LPAR management time for both CF LPs and OS/390 LPs that use ICMF. LPAR management time for a CF LP is significant and can be high.

Information APAR II09294 contains considerations for CFs that run in shared LPs:

- Recommendations for CFs with CFR Channels (non-ICMF).
- Recommendations for CFs that use the Integrated Coupling Migration Facility (ICMF).
- The CF weight for the CF logical CP should be greater than the weight per logical CP for MVS.

A good source of information is found in the WSC Flash 9609 *MVS/ESA Parallel Sysplex Performance LPAR Performance Considerations For Parallel Sysplex Environments*, OZSG023608.

1.4 Capacity Planning and I/O

In the following sections we explain the concepts:

- Relative I/O Content (RIOC)
- Relative Lock Content (RLKC)

1.4.1 Relative I/O Content (RIOC)

The Relative I/O Content (RIOC) is an entity often used and referred to in CP2000.

RIOC is the ratio of the total number of DASD I/O operations per second to the total CEC power for the same time period. It provides a way of describing a workload in terms of I/O content where a higher number shows a workload of

higher I/O content. RIOC is used to calculate the number of I/Os an application creates when its usage of CEC power (M-value) is known.

CP2000 calculates the RIOC by the following formula:

RIOC Definition

$$RIOC = \frac{S}{(M\text{-}value \times B)}$$

where:

- M-value is the power of the CEC,
- B (Busy) is the CPU utilization as a fraction,
- S is the number of I/Os per second.

Relative I/O Content varies by application type software, so typical measured systems show different values for RIOC. TSO will be different from IMS/DB2, CICS/VSAM from CICS IMS/DB, and so forth.

Some observations concerning RIOC[2] are:

- Even if you change the CPU power, but the workload is unchanged, then the RIOC remains constant.
- If the workload is changed, then the RIOC changes.
- RIOC has been observed to be steadily decreasing year after year (at a rate of nearly 10%) as software and applications have exploited larger central and expanded storage via the Data in Memory techniques, and also because the transactions use more CPU.

CP2000 makes the assumption that the RIOC of a workload remains constant over time, thus implying that the software and the application design are unchanged. Some factors which may change, and therefore affect RIOC values, are the system control program, microcode, database subsystems, and application tuning such as buffering and application characteristics.

Note: Some Database subsystems such as DB2 use memory buffering to resolve most I/Os. When physical I/Os take place to DASD devices they are chained. Because of this design, it is sometimes difficult to relate the number of I/Os to particular applications or workloads.

It is important to review the RIOC of both the current and future workloads because the DASD I/O content of the future workload is a key factor in deciding both CEC and DASD requirements. If the workload you are studying is one with which you are not familiar, you may not be able to assess the validity of the RIOC and therefore, you should accept the system RIOC. The system RIOC is the total system I/Os divided by the total M-value for the entire system and the total CPU consumption.

Typical RIOC values depend on many factors. Based on data from the CP2000 development data base (1997) the all average RIOC is about 0.26. Smaller processors tend to have higher RIOCs, and larger processors have lower RIOCs. The 10th and 90th percentile are 0.01 and 0.40 respectively. Table 2 on page 33 gives you some guidance about typical application RIOCs.

[2] For further discussion of RIOC, you should refer to the IBM Systems Journal, Vol. 20, No. 1, *Processor, I/O Path and DASD Configuration Capacity*.

Table 2. Typical Non-Sysplex RIOCs for Different Workloads (1997)		
Workload type	**RIOC**	**Remark**
Long-running batch	0.12 - 0.25	
Short-running batch	0.12 - 0.20	
Light CICS transactions	0.10 - 0.20	lower in the DB2, DIM and MRO environment
Heavy CICS transactions	0.12 - 0.25	lower in the DB2, DIM and MRO environment
IMS transactions	0.15 - 0.30	lower in the DB2, and DIM environment
Normal TSO	0.06 - 0.16	
Heavy TSO	0.12 - 0.20	
TSO using OS/390 UNIX Services	0.08 - 0.35	Mixed OMVS workload with TSO OS/390 UNIX Services usage, webserver, file transfer

RIOC in Parallel Sysplex

You can expect to see the value for RIOC decrease slightly after migrating to a Parallel Sysplex environment. The Parallel Sysplex RIOC depends on the particular subsystem implementation. This depends on buffer invalidation (which may cause extra I/Os) and on XES services overhead (causing an increase in CPU consumption).

You could see the S value (number of I/O per second) increasing for an IMS/DB environment because of the Cross Invalidate (XI) effect; it will remain almost the same in a DB2 and CICS/VSAM RLS environment due to the different type of data sharing implementation. At the same time we experience a slight increase of the (M-value x B) value due to the coupling overhead.

The following formulas can serve you as rough guidelines:

- $DB2\ RIOC_{Parallel\ Sysplex} = 0.75\text{-}0.90 \times DB2\ RIOC_{Single\ System}$
- $IMS/DB\ RIOC_{Parallel\ Sysplex} = 0.80\text{-}1.00 \times IMS/DB\ RIOC_{Single\ System}$

Note: The RIOC values described above is a function of the HW Configuration (number and types of CECs and CFs) and the workloads (number of systems and data sharing percentage).

The *System RIOC* values typically will decrease less than 10% in a Parallel Sysplex.

All these effects will be discussed in detail in Chapter 2, “Capacity Planning in Parallel Sysplex” on page 45.

1.4.2 SRM I/O Service Units

Systems Resources Manager (SRM) uses the service definition coefficients to assign additional weight to one type of service relative to another, allowing the installation to specify which type of resource consumption should be emphasized in the calculation of service rates. For example, an IPS may contain the following service definition coefficients:

```
CPU=10.0    IOC=5.0    MSO=0.0    SRB=10.0
```

Let us take a quick look at the different categories of service units, for more information refer to *OS/390 MVS Initialization and Tuning Guide*, SC28-1751.

- **CPU service units**. Refer to 1.3.6, “SRM CPU Service Units” on page 29.
- **SRB service units**. Refer to 1.3.6, “SRM CPU Service Units” on page 29.
- **I/O service units** relate to measurement of individual data set I/O activity and JES spool reads and writes for all data sets associated with the address space. SRM calculates I/O service using either I/O block (EXCP) counts or device connect time (DCTI), as specified on the IOSRVC keyword in the IEAIPSxx parmlib member. If DCTI is used to calculate I/O service, operations to VIO data sets and to services that the channel measurement facility does not time are not included in the I/O service total.

 When an address space executes in cross memory mode (that is, during either secondary addressing mode or a cross memory call), the EXCP counts or the DCTI will be included in the I/O service total. This I/O service is not counted for the address space that is the target of the cross memory reference.
- **Storage service units**. Refer to 1.5.4, “SRM Storage Service Units” on page 38.

1.4.3 Relative Lock Content (RLKC)

IRLM uses system locking services to support IMS/DB and DB2 data sharing. The OS/390 Cross System Extended Services (XES) handles high-performance multisystem locking and caching with integrity and consistency across data-sharing systems.

As with I/O rates, locking rates vary by data manager, workload, operating system level and CF microcode level. The *Relative Lock Content* (RLKC) for a workload is analogous to the RIOC of a workload. The RLKC is the ratio of the total number of lock and unlock requests per second toward the CF to the OS/390 CPU power consumed for some time period.

RLKC Definition

The following formula shows you how to calculate the RLKC:

$$RLKC = \frac{L}{(M\text{-}value \times B)}$$

where:

- L represents the total number of lock and unlock requests per second in the measured period.
- M-value represents the total power of CEC.
- B represents the utilization of the workload sharing data.

Note: How do you estimate the value of B? That is defined in some detail in the next chapter.

Typical (1997) Non-Sysplex DB2 RLKC number

For DB2 V5 a RLKC value of 0.33 can be used to project locking when a Type 2 Index is used. The range for DB2 V5 RLKC values will typically vary from 0.25 to 0.50.

Typical (1996) Non-Sysplex IMS RLKC number

Data from both customer and internal benchmarks has shown that for IMS OLTP workloads, a RLKC value of 0.25 is reasonable. The range for IMS OLTP is from 0.15 to 0.30.

The lock requests are a subset of all requests to the CF. How to obtain an estimate of the RLKC in your environment is further discussed in:

2.2, “DB2 Parallel Sysplex Data Sharing Considerations” on page 63.
2.3, “IMS/DB Parallel Sysplex Data Sharing Considerations” on page 74.
2.5, “CICS/VSAM RLS Data Sharing Considerations” on page 84.

1.5 Capacity Planning and Storage

Storage is often no longer a critical resource for a capacity planning study. This is because of its inexpensive price combined with the fact that there sometimes are big increments between two consecutive storage sizes. Thus you often “over-configure” storage by a fair amount.

In the following topic, we will briefly discuss how to make storage capacity planning and how to assess how much expanded storage you need.

1.5.1 Central Storage Capacity Planning

If you want to carry out a detailed capacity planning study for central storage, the steps are pretty much the same as they are for CPU capacity planning. CP90 offers a simple approach based on the assumption that there is a correlation between the CPU resources and the corresponding central storage. This is discussed in a little more detail in 6.4.2, “Storage Projections” on page 192.

Parallel Sysplex offers significant benefits in data sharing environments. Since buffer pools now become a Parallel Sysplex entity, capacity planning needs to take into consideration sizing and placement of buffer pools. This is discussed further in:

2.2.5, “DB2 Buffer Pool Sizing in Parallel Sysplex” on page 70
2.3.5, “IMS/DB Buffer Pools Sizing in Parallel Sysplex” on page 80

1.5.2 Parallel Sysplex Related Storage

If you are not already running in a Sysplex, the OS/390 system storage for each system is approximately the same as it would have been for one large OS/390 system, but you should allow for the following additional central storage:[3]

- XCF storage - around 6 MB
- Console storage - around 3 MB
- RMF SMF data buffer - around 32 MB[4]
- JESXCF storage depending on the number of JESXCF members
- IRLM needs additional storage for P-locks (see information APAR II09146)

[3] Some of the following storage is “virtual,” thus not necessarily adding to the total working set.

[4] User specified in the interval 1 MB - 2 GB. Not all of the buffer is available for SMF records. 64 bytes per SMF record is used for a directory entry.

These numbers are based on early experiences and are approximate. If you are already running in a Sysplex, you do not need to allow for all of this additional storage.

1.5.3 Central and Expanded Storage Considerations

This section discusses the different types of storage, CS (central storage) and ES (expanded or extended storage).

Central Storage Versus Expanded Storage Rule Of Thumbs for 9672

IBM 9672s and 9674s have one type of storage only, which can be divided into central and expanded storage. Storages size is defined at power-on time.

The general recommendation is to define as much as possible of the 9672 storage as central. If special considerations apply, such as the exceptions listed below, then you may define part of the storage as expanded. These exceptions apply if you have implemented certain functions that have a *requirement* for expanded storage.

On IBM 9674s only define expanded storage if a CF in that 9674 requires over 2GB storage. This simplifies storage planning for the CF.

In 9021 based CECs, where CS and ES storage is physically different, and the cost is different, there are different sizing rules of thumb. Some of these rules of thumb were based on historical observations and should not be looked at as general rules. You may have good reasons to configure your system differently. For example, some rules of thumb with respect to CS and ES on CECs are:

Central Storage Versus Expanded Storage Rule Of Thumbs for CECs

- Base your rule of thumb on paging rates. Include available CPU capacity, storage, price of CPU capacity, price of storage and response time considerations in your evaluation:
 - When CS-ES traffic is 1000 pages/sec per CP in 9021 H5 CECs, you spend about 3% of CPU power for page movements.
 - When CS-ES traffic is 1000 pages/sec per CP in 9672 R5 based CECs, you spend about 3% of CPU power for page movements.
- The time spent to resolve a page fault on certain CECs are shown below. These are typical numbers and includes both software and hardware time:
 - 30 µ-seconds (9021 based CECs)
 - 90 µ-seconds (9672 R2 based CECs)
 - 40 µ-seconds (9672 R4 based CECs, estimated)
 - 30 µ-seconds (9672 R5 based CECs, estimated)

Specific Functions That May Need Expanded Storage?

Expanded storage is used for hiperspaces and paging/swapping. In the IBM 9672 system you should define as much as possible of its processor storage as central storage (in other words up to 2 GB per OS/390 Image).

In the "old days" the recommendation to configure all storage as central storage was mostly true for non-swappable type workloads. Often people liked to define a little expanded storage to enable the system to handle and especially warn you that storage is starting to be constrained. If all storage was defined as central you could have a system working well which very quickly became badly performing when storage began to be constrained. As is always the case in capacity planning and performance management, this depends on a lot of factors one of them being how well you track system imbalances before they become a real problem. *Today the recommendation for 9672 CECs is to define as much as possible of the storage as central storage*.

The following lists *exceptions* for the previous recommendation, functions that may need expanded storage:

- **VSAM hiperspaces**

 VSAM hiperspaces can be used directly, or via CICS, IMS, Batch LSR, NetView, and so forth. VSAM buffer pools do not absolutely require hiperspaces or expanded storage.

- **Hiperbatch**

 Hiperbatch absolutely requires expanded storage.

- **DB2 hiperpools**

 On 9672 processors DB2 hiperpools should only be used if the total size of the buffer pools in a single DB2 group member is larger than about 1.6 GB. This is not likely in a data-sharing environment. The restriction comes from the maximum size of a single DB2 DBM address space which is 2 GB. DB2 does not use data spaces.

- **DFSORT Hipersorting**

 For DFSORT Hipersorting, ES is highly recommended; about 1/5th at least should be defined as ES. This is further discussed in *Expanded Storage is Still Very Important for DFSORT*, WSC Flash 9510.

- **PDSE**

 PDSE uses hiperspaces for member-level caching, for example. PDSE does not absolutely require hiperspaces or expanded storage.

- **IMS Virtual Fetch**

 IMS Virtual Fetch requires expanded storage.

- **Paging/swapping**

 Paging/swapping does not have an absolute requirement for expanded storage.

- **Virtual I/O (VIO)**

 Expanded storage can be used for VIO pages for any VIO job in a system with no VIO journaling, or a non-journaled VIO job regardless of whether VIO journaling is active on the system. Expanded storage is only used for VIO when no significant increase in demand paging is the result.

The general conclusion here could be that only hiperbatch requires expanded storage and that DFSORT performance is better if expanded storage is defined.

When there are many swappable address spaces (TSO and batch), SRM′s IEAOPT parameters may need adjustment to avoid unnecessary physical swapping. With ES, physical swaps will go to ES if feasible. But in the CS-only case, the user will be physically swapped to DASD, if his think time is longer than the system think time. This can be avoided by increasing the system think time high threshold in the LSCTMTE parameter, which has a default value of 5 seconds, which is a very low value. The other parameter, LSCTUCT, which is used to adjust the system think time, can be left as its default value (20,30). In goal mode LSCTUCT is not used. The LSCTMTE parameter, which has the default of LSCTMTE=(0,5), could be for example

```
LSCTMTE=(0,333)
```

Appendix B, “Processor Storage Management and Extended Storage” on page 285 contains an explanation of why defining some ES can improve OS/390 storage management.

1.5.4 SRM Storage Service Units

Systems Resources Manager (SRM) uses the service definition coefficients to assign additional weight to one type of service relative to another, allowing the installation to specify which type of resource consumption should be emphasized in the calculation of service rates. For example, an IPS may contain the following service definition coefficients:

```
CPU=10.0    IOC=5.0    MSO=0.0    SRB=10.0
```

Let us take a quick look at the different categories of service units, for more information refer to *OS/390 MVS Initialization and Tuning Guide*, SC28-1751.

- **CPU service units**. Refer to 1.3.6, “SRM CPU Service Units” on page 29.
- **SRB service units**. Refer to 1.3.6, “SRM CPU Service Units” on page 29.
- **I/O service units**. Refer to 1.4.2, “SRM I/O Service Units” on page 33.
- **Storage service units** are calculated using the formula:

 (*central page frames*) *x* (*CPU service units*) *x* 1/50

 Where 1/50 is a scaling factor designed to bring the storage service component in line with the CPU component. Not included in the storage service unit calculation are the central storage page frames used by an address space while referencing the private virtual storage of another address space through a cross memory function (that is, through secondary addressing or a cross memory call). These frames are counted for the address space whose virtual storage is being referenced. Expanded storage pages are not included in the storage service unit calculations.

The working set size of 50 pages, used in the previous calculation, is very small compared to today′s working sets. In other words, MSO is not very useful and we recommend setting

```
MSO=0.0
```

1.6 Capacity Planning Activities

Capacity planning is a subset of the systems management discipline *performance management*. Performance management is the activity that monitors and allocates data processing resources to applications according to *service level agreements* or equivalent. This section discusses the steps you need to carry out to do capacity planning.

Capacity planning activities include and are linked to the other processes of performance management and systems management disciplines. Before we can conduct a capacity planning study, we have to define our activities and especially the objectives. The following list shows the major activities:

1. Select Capacity Planning Method
2. Measure Current Workload
3. Check for a Balanced System
4. Predict the Future
5. Model and Interpret Data
6. Collect and Document Capacity Planning Results
7. Communicate the Capacity Planning Results

How Often Do You Do Capacity Planning?

Depending on the selected capacity planning method and tool, the number of capacity planning runs per year can span between one and 12. The capacity planning period should not exceed 24 months, unless there are very good reasons for this, such as a stable workload.

Between consecutive capacity planning runs, complete reporting is done. This ensures adequate time to adapt to exceptional changes or deviations from the forecast. Figure 10 on page 40 provides an overview of the steps in a capacity planning study.

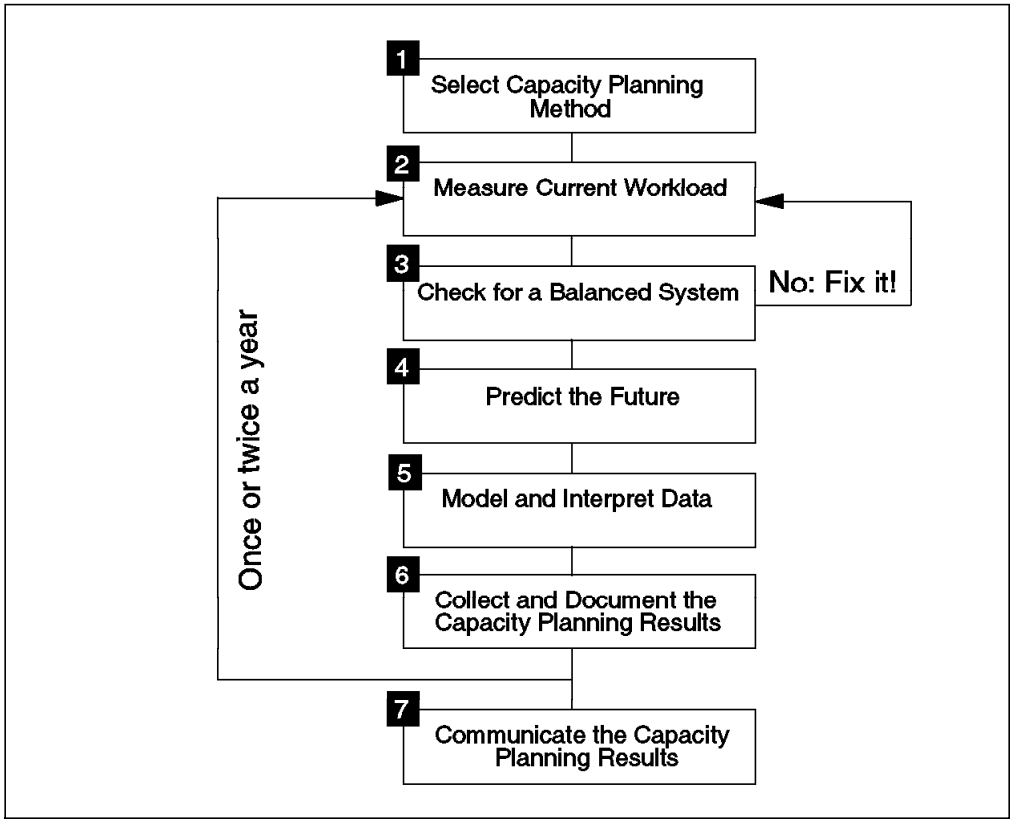


Figure 10. Capacity Planning Activities - Step-by-Step

Each of these steps are described in some detail in the following text.

1.6.1 Decide Capacity Planning Method



Based on accuracy of the projections, the first step in a capacity planning study is the choice of method. Several capacity planning methods are outlined in 1.2, “Capacity Planning Methods” on page 4. According to the selected choice, you collect different data from multiple sources of the installation.

Let us assume that we choose CP2000 or CP2000 due to its ease of use and great flexibility.

1.6.2 Measure Current Workload



Standard performance measurement tools are typically used to capture resource usage that is currently required for the workload. The measurements should reflect the activity by unit of work (transaction, batch job, and so forth). Transactions can be grouped at a business workload level as projections are often discussed at that level.

The question about how often an installation should do capacity planning involves the following:

- Time for the analysis
- Time for approving the investments
- Time to carry out

Dimensioning Workload

Now, there should be an understanding of what workload cannot be delayed, that the business depends on, or that is essential for the daily business. This workload is called the *dimensioning workload*. There should be a good understanding of what the dimensioning workload is composed of and how it is measured.

Business Elements

Business Importance

Often, the total workload contains several business elements. The dimensioning workload may be only a part of the total workload. Each of the business elements might have a different *business importance* to the company. Examples of business elements are Automatic Teller Machine (ATM) workload (CICS), critical day time batch and Information Center (TSO). Most operating systems have measuring mechanisms that provide grouping facilities to report resource consumption for different business elements.

In OS/390 (MVS), this is possible due to the Performance Group Number (PGN) and the associated Reporting Performance Group Number (RPGN). For OS/390 (MVS/ESA V5.1 and upwards) systems operating in *Goal Mode*, these aspects change with the introduction of the workload manager (WLM) constructs: workloads and service classes, reporting classes.

Peak Hour?

Peak Day?

Peak Week?

Peak Month?

Often, a business has *peaks* in their system workload at certain points in time. This depends very much on the business. A bank, for example, might experience a *peak hour* every day between 3 and 4 p.m. Then they might also experience a *peak day* every Monday because of long opening hours at the branches. They may also experience *peak weeks* every last week in the month due to salary and other monetary transfers. The bank also knows every December they experience a higher workload than compared to other months. Measurements should be taken to define *when* the dimensioning workload you are planning for is running.

Spare Capacity?

Also, continuous availability aspects should be taken into account. This means that maybe it is wise to over-configure parts of the system. The reason is that part of the system may need to be taken down for service on a regular basis, primarily due to planned maintenance but also because of other outages. This will affect the SDP.

Contingency Allowances

Contingency allowances should also be applied. In a steady state environment, this allowance is very small and grows with the fluctuation of the workload. The SDP is used to take this into account.

Note: In a Parallel Sysplex, the peaks on several systems can, to an extent, level each other out due to dynamic workload balancing. This effect provides a higher effective capacity in certain Parallel Sysplex environments. Many planned and unplanned outages also need less spare capacity in the Parallel Sysplex than in the non-Parallel Sysplex environment.

Adjust your SDP to accommodate for the effects of dynamic and static workload balancing in the Parallel Sysplex. For a definition of SDP, refer to 1.3.4, "Peak-to-Average Ratio (PAR) and Saturation Design Point (SDP)" on page 21.

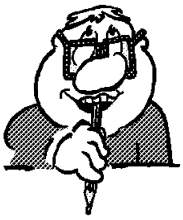
1.6.3 Check for a Balanced System



Perform an analysis of the captured data to ensure data validity. At this point, check whether you have a balanced system or not. Serious out-of-balance symptoms might appear, such as service level agreements that are no longer met or parts of the systems showing stress symptoms.

For out-of-balance (over-utilized) systems, latent demand might be a problem when trying to estimate the workload. For example, CICS should not be going short on storage; IMS regular message region occupancy should be less than, say, 65%; average DASD response time should not exceed, say, 10 ms; OS/390 (MVS/ESA) capture ratio (CR) should not be below, say, 0.85, and so forth. *Care* should be taken at this point. The best solution is probably to *fix the problem* and then rerun the measurements on your system.

1.6.4 Predict the Future



Now comes the hard part: assessing the future. It is very important to know the growth rate of your business and how to translate this into growth in resources. At this point, you must decide for which period you want to plan. Planning periods often go from 12 to 24 months. It is not advisable to expand the planning period because the impact of unexpected business decisions is too high. This causes a reduction of planning accuracy and, therefore, a reduction in the validity of the study.

With the decision to run "what if" sessions, the time comes when all the right people are consulted and asked questions such as:

When?
Why?
What?
Where?
How Much?
How Many?
How Long?

- What are the company plans and goals?
- How many people will work in department XYZ next year?
- Will you change critical business processes in the future?
- How are you converting business processes to capacity planning input?
- Are you changing the IT strategy?
- Are you migrating to another IT architecture?
- Are any business mergers taking place soon?
- Do you plan to develop or buy a new application?
- Do you plan access to the World Wide Web?
- Are you implementing more Client/Server based applications?
- Are the users changing their usage of the system?

The new or changed business processes have to be converted and translated in terms of transactions, associated CPU, I/O, and storage resources required to support them both in terms of the CEC and the CF. Even though this is the tough part, it is also the fun part, as you get to know your business very well. If this part of the capacity planning study is not carried out with the proper level of detail, you might get very uncertain (or even wrong) predictions. Remember that well-defined and implemented systems management disciplines increase the accuracy of the predictions. *The old adage, garbage in, garbage out, still holds true.*

1.6.5 Model and Interpret Data



Now we are going to model the future workload to predict how it runs on various software and hardware configurations. Several growth scenarios are often a product of this step. Several techniques can be applied in this step. Depending on accuracy, detail, the invested time, money, and available skills, the appropriate method and tool is chosen. See also the discussion in 1.2, “Capacity Planning Methods” on page 4 for the relationship of accuracy, effort and cost. Special emphasis on modelling the transition to a Parallel Sysplex is discussed in Chapter 2, “Capacity Planning in Parallel Sysplex” on page 45.

1.6.6 Collect and Document the Capacity Planning Results

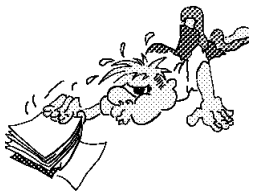


The last step is to *document* the results of the capacity planning study. A report showing the effect of various growth scenarios addressing the business objectives is generated. Now, it is a good idea to show the relationship between the last time′s prediction and the actual workload usage at the point of measurement. Probably now (or the next time around) it is a good idea to provide *feedback* to the people who provided information about the future: how accurate were their predictions? Did the business evolve the way it was planned, and so on. The feedback might ensure a more accurate prediction next time and motivate the information providers so they can see the value of their information. To do sound capacity planning, the steps must be repeated over and over again.

How often depends on the details provided and the characteristics of the business. One or two times a year is a fair rule of thumb. When doing capacity planning as a continuous process, most installations find it easier and easier as they get more and more experienced.

The result of the capacity planning study for a Parallel Sysplex outlines both vertical and horizontal growth scenarios.

1.6.7 Communicate the Capacity Planning Results



Running successfully through all the previous activities is worthless if the results are not communicated to the responsible people. To communicate the results, establishing communication links to the management, financial, and operating people is highly recommended.

This also makes life a lot easier the next time around.

Chapter 2. Capacity Planning in Parallel Sysplex

At a Glance

This chapter presents the Parallel Sysplex architecture, its components, and the issues related to capacity planning in a Parallel Sysplex environment.

Coupling technology, data integrity, workload balancing and data sharing considerations for DB2, IMS/DB, and CICS/VSAM RLS are outlined with the emphasis on capacity planning.

Recommended Reading:

- *DB2 for OS/390 V5 Data Sharing: Planning and Administration*, SC26-8961.
- *IMS/ESA V6 Administration Guide: System*, SC26-8730
- *OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook*, SG24-2076
- *S/390 MVS Parallel Sysplex Performance*, SG24-4356

© Copyright IBM Corp. 1998

2.1 An Extension to the S/390 Architecture

This section gives an overview of the Parallel Sysplex environment. We briefly discuss hardware and software and look at the issues that need to be considered for capacity planning purposes.

Single Image for Operations, Systems, and Applications

The Parallel Sysplex environment permits a workload on an ES/9000 or S/390 Parallel Enterprise Server to grow from a single system to a complex of a maximum of 32 systems that appear as a single, powerful image to the S/390 user. For the largest and most demanding customers, the Parallel Sysplex provides almost twenty five times the power of today′s largest commercial IBM mainframe.

A single image complex may contain one or more copies of operating systems, but with the human perception of only one system to manage for everyone including operators, schedulers, system programmers, performance analysts, application, developers and users.

Continuous Availability

Through its parallel architecture, Parallel Sysplex establishes an opportunity for providing continuous availability. Parallel processing allows several images of the OS/390 operating system to process work concurrently. The kind of work best suited for processing in a Parallel Sysplex is work that is divided into small units. A good example is work managed by on-line transaction processing (OLTP) programs. Because of the “shared-all” architecture, Parallel Sysplex exploits a near continuous computing environment for availability and growth. Therefore applications that need high availability and continuous operation are good candidates for Parallel Sysplex processing.

Your S/390 Investment Preserved

The Parallel Sysplex is based on the S/390 architecture, a compatible extension to the existing ESA/390 architecture. This innovative architecture provides a set of parallel coupling and data sharing facilities that provide a flexible processing platform exploiting the latest cost/performance improvements of microprocessor technology and the performance advantage of parallel processing. These parallel systems are compatible with today′s existing S/390 applications. This compatibility means that:

- Extensive business investment in software, hardware and skills is preserved.
- Current applications benefit from the additional value provided by Parallel Sysplex, including availability, scalability, and using CMOS-based CECs, a more granular growth option.
- ES/9000 9121 511-based and 9021 711-based models can be integrated into the Parallel Sysplex architecture.
- The Parallel Sysplex is the base for the enhancement of current and new systems functions.

Parallel Sysplex Provides Good Scalability Compared to the Multiprocessor Architecture

Scalability in a Parallel Sysplex architecture means that it is not subject to the same "drop off" in benefit from adding more images as a tightly coupled multiprocessing (MP) CEC is when more CPs are added. As more OS/390 images are added to the Parallel Sysplex, you achieve almost *linear growth,* as shown graphically in Figure 11 on page 47.

With the *Multiprocessor* curve, the point of diminishing return occurs rather quickly because of hardware and software design limitations (MP ratio considerations). Do not expect to see more than about 10 to 12 CPs in a tightly coupled multiprocessor.

The *Parallel Sysplex* line shows that the design of the Parallel Sysplex architecture is such that one additional OS/390 image (up to 32) has an impact of less than *half a percent* of degradation with 100% data sharing. This is in addition to the *enablement* cost for data sharing in a Parallel Sysplex. The total cost of Parallel Sysplex is discussed further in for example 2.2.3, "Parallel Sysplex CPU Cost for DB2 Data Sharing" on page 65.

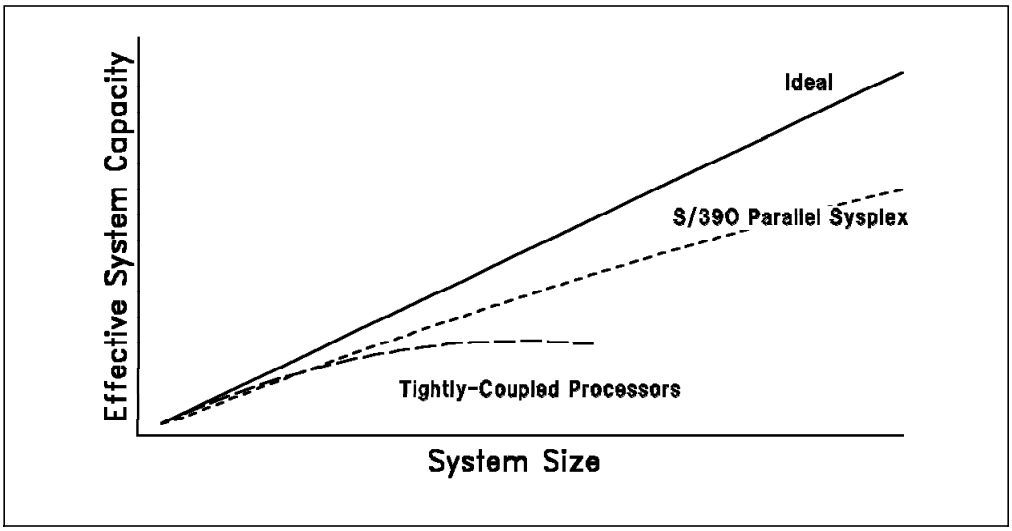


Figure 11. Scalability in the Parallel Sysplex

Low Total Cost Of Computing

One major objective of the Parallel Sysplex architecture is to improve the total cost of computing (hardware, software and systems management) for large commercial processing systems. The S/390 Parallel Enterprise Servers provide a platform that is competitive with alternatives. This, together with the Parallel Sysplex software pricing, provides total system capability for many aspects of businesses' commercial processing needs at a competitive price.

The Parallel Sysplex provides continuous availability, thus reducing business costs. In an era when the cost of labor is ever increasing, systems management capabilities and continuous availability features are key to future success.

Three Parallel Sysplex Challenges

To face all these requirements, the Parallel Sysplex architecture must solve the three following challenges:

1. *Share data* among all the involved OS/390 systems with integrity and performance. This is done with the assistance of hardware microcoded functions, the CF and coupling links. Parallel Sysplex data sharing involves OS/390 and the database manager subsystems.
2. *Balance the workload* among all the systems that are allowed to run the application. This is done through dynamic workload management that involves OS/390, VTAM and the transaction manager subsystems.
3. Provide a *single image view* for systems management. Capacity planning is one of the components of systems management.

Parallel Sysplex Benefits

To summarize S/390 and Parallel Sysplex provides:

- **A**daptive workload management
- **B**usiness competitive advantage
- **C**ontinuos secure access to critical business on demand:
 - At lower incremental cost
 - At lower business risk
 - And you are unlikely to outgrow it

2.1.1 Workload Management

Dynamic Workload Balancing

Parallel processing in a Parallel Sysplex is the ability to simultaneously process a mixture of work with various characteristics across many images, without incurring changes or inhibiting access to data. This work is defined in units of work, each one having its own characteristics in terms of performance goals and business importance.

Dynamic workload balancing can be used to spread the units of work according to the processor load to achieve response time in accordance with goals. This allows all units of work to get access to all the shared data, thus utilizing resources more efficiently. The OS/390 and MVS/ESA V5 workload manager function and specific levels of the data base transaction and transaction manager subsystems are involved in this dynamic process.

For capacity planning, dynamic workload balancing allows more flexibility and granularity resource usage. That means the CICS, IMS, and DB2 transactions can now be monitored and balanced at the transaction level. Monitoring and reporting tools provide details of delays and usage, thus offering a better understanding of the resource consumption.

If your installation today is serving workloads running on several CECs, then Parallel Sysplex offers you the opportunity to use spare capacity on one or more CECs by levelling the workload across a subset of the Parallel Sysplex or across the entire Parallel Sysplex.

2.1.2 Data Integrity and Buffer Pool Coherency Considerations

Data integrity is an important issue when sharing data. This is also the case in a Parallel Sysplex. Let us first briefly discuss data integrity in a *single OS/390 image,* followed by *several OS/390 images without Parallel Sysplex* and conclude the section with a discussion of data integrity in a *Parallel Sysplex.* This discussion is provided to aid later discussions related to Parallel Sysplex capacity planning with CP2000.

2.1.2.1 Data Integrity before the Parallel Sysplex: One System

Data Integrity in One OS/390

When only one OS/390 system has access to database records, current S/390 data management products are able to address data integrity for thousands of active users. They apply a *data base manager locking* technique to ensure data is valid for multiple tasks that may be accessing the same data. They use in-storage buffering to improve transaction performance, holding frequently used data and recently used data in buffer pools to avoid the overhead of reading data from DASD.

You may have different reasons to grow from one OS/390 to multiple OS/390 systems, for example:

- Demand for continuous availability
- Workload has grown too large for one OS/390 system
- Lower overall cost of computing

2.1.2.2 Data Integrity before the Parallel Sysplex: Several Systems

Multisystem Environment

In a multisystem environment, it is more complex to do locking and buffering than in a single system. When two or more systems need to share data, communication between those systems is essential to ensure that the data is consistent and that each has access to the data.

One way to share data between systems is through *reserve/release.* This sharing is accomplished through a hardware feature of the DASD control unit together with the reserve/release function of the GRS function of the operating system. This reserve/release serialization method has severe performance implications for anything but limited access to data.

Another way to serialize data is through *message passing*.

Let us assume two OS/390 images, S1 and S2, share the same database, as shown in Figure 12 on page 50. When our discussion starts, the same record ABC was stored in each buffer pool in each OS/390 image.

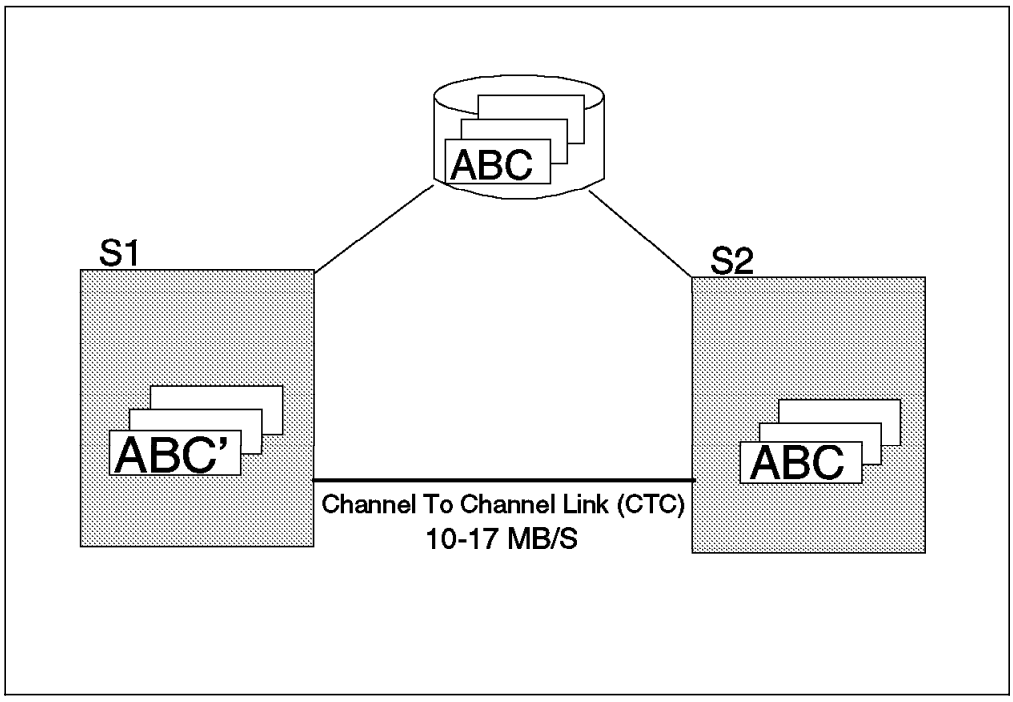


Figure 12. Multisystem Data-Sharing

An Expensive Communication Mechanism

S1 needs to update the ABC record. To obtain a lock, S1 sends a message across a channel-to-channel (CTC) link to S2 requesting access to the record. If S2 is not using the record, it sends a message back to confirm that S1 can have the access it requires. Consider two things about what was just described:

1. Both S1 and S2 applications had to stop what they were doing to communicate with each other.
2. If there are three or more systems, this scheme becomes even more complex. The *cost* of managing the sharing in this method is *directly proportional* to the number of systems involved.

This problem is even more complicated than just described. Let us suppose that S2 wants to read the ABC data and that the most recent copy of the record ABC has not yet been written to the DASD that both systems are sharing. It might be in an internal buffer of S1 (record ABC'). In this case, S2 must first be informed that the copy on the disk is not valid. S1 must then write it out before S2 can have access to it.

Buffer Pool Coherency

A mechanism that tells S2 whether the copy of the data in its buffer pool is the most recent one is said to *maintain buffer pool coherency*.

Some data base managers offered specific data sharing mechanisms before the Parallel Sysplex became available. IMS/DB provides 2-way data sharing through IRLM (using VTAM CTCs for communication), as an example of this implementation. Since this data sharing scheme is expensive both in terms of time and CP resources, it is not widely used.

2.1.2.3 Data Integrity in Parallel Sysplex

Simple, Flexible, and Inexpensive Data Sharing

Parallel Sysplex provides, among other functions, the ability to manage database integrity (locking and buffering) for up to 32 OS/390 images in a much simpler way than in a non-Parallel Sysplex multisystem configuration. We need to allow the sharing of data among *many systems* without causing the problems described in the prior example.

Data must be shared in such a way that adding systems does not increase the complexity or overhead, while still preserving data integrity across updates. To solve this complicated problem, a new sharing technique is needed. This is implemented through the CF technology that manages the *communication* and *serialization* necessary for *sharing* data. The CF function works with *structures* put in the CF. These structures help manage the serialization of access to data and coherency of the data.

OS/390 (MVS) has been enhanced to provide a set of services, the XES services, that enable subsystems and authorized applications to use the CF structures. These services provide buffer pool coherency, and locking serialization that allow data sharing to occur between many OS/390 systems. As an example of how this works, let us look at Figure 13, which shows an example of how cache structures are used when two systems are sharing data.

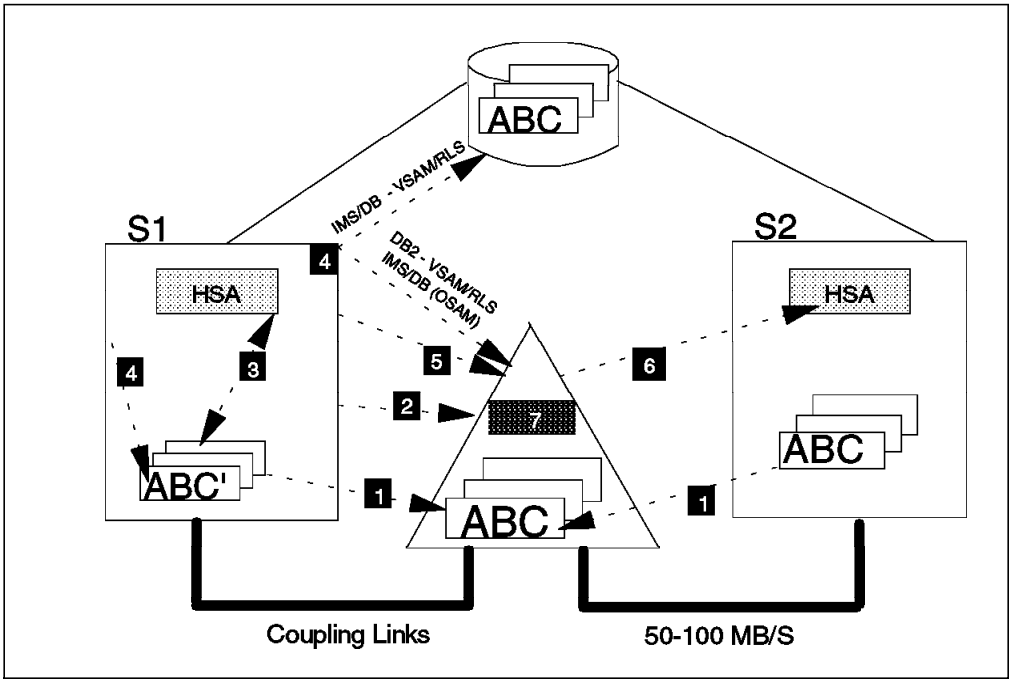


Figure 13. Multisystem Data Sharing in Parallel Sysplex

Global Buffer Directory in the CF

Looking at the local buffer pool for each system, you can see some records (including the ABC record). Let us assume both systems have read record ABC in the past, and a valid current copy exists in each buffer pool.

1 Both systems have registered their interest in record ABC with the CF. Now, an application in S1, on the left, needs to update this ABC record as ABC'. **2** The DBM in S1 invokes an OS/390 service that calls the CF to obtain an exclusive lock for the update. Depending on the subsystem implementation, the

lock may be obtained for one or more records. Let's imagine that the lock was granted.

3 Now S1 looks in its HSA vector bit table associated with the buffer to see if the record in the buffer is a valid copy. Assume it is valid, so S1 does not have to go further (to DASD or CF) to obtain a copy.

4 S1 changes the record to ABC′. It is changed in S1's local buffer, and depending on the subsystem protocol, it is written to:

- A cache structure of the Coupling Facility. An example is DB2 or IMS DB V6 (OSAM) using a store-in technique.
- A cache structure in the Coupling Facility *and* to DASD. Examples are VSAM/RLS and IMS DB V5 (OSAM) using a store-through technique.
- DASD only. An example is IMS DB V5 (VSAM) using a directory-only technique.

5 A signal is sent by the database manager to the Coupling Facility to show that this record has been updated. The CF has a registration in a cache structure directory of every system in the Parallel Sysplex that had a copy of the record ABC, which has now been changed to ABC′. A directory entry is used by the CF to determine where to send cross-invalidation signals when a page of data is changed or when that directory entry must be reused.

6 Without interrupting any of the other systems in the Parallel Sysplex, the CF invalidates all of the appropriate local buffers by changing the bit setting in the HSA vector associated with the record to show that the record ABC in the local buffer is down level. This operation is called *cross-invalidation*. Buffer invalidation may also occur under other circumstances, such as when a contention situation is resolved through directory reclaim.

7 At this step, the serialization on the data block is released when the update operation is complete. This is done with the use of global buffer directories, placed in the CF, that keep the names of the systems that have an interest in the data that reside in the CF. This is sometimes referred as the *unlock operation*.

In a store-in operation, the database manager occasionally initiates a cast-out process that off-loads the data from the Coupling Facility to the DASD.

The next time the images involved in data sharing need record ABC, they know they must get a fresh copy (apart from S1, which still has a valid copy in its local buffer). The images are not interrupted during the buffer invalidation. When the lock is freed, all the images correctly reflect the status of the buffer contents.

If S1 does not have a valid copy in the local buffer (as was assumed in step **3**), it must read the record from either the cache structure in the CF or from DASD. The CF only sends messages to systems that have a registered interest in the data, and it does not generate an interrupt to tell each image that the buffer is now invalid.

Share Locks

This example details the steps necessary every time a record is referenced. Note that the shared locking also takes place if a system reads a record without ever doing an update. This guarantees that no other images update the record at the same time. The example discussed here can easily be expanded to more than two systems.

To summarize, the following is the sequence of events for accessing a database with integrity in a Parallel Sysplex:

1. An element is locked before it is updated.
2. An element is written to DASD no later than sync point or commit time.
3. Buffer invalidation occurs after the record is written.
4. Locks are released during sync point or commit processing, and after all writes and buffer invalidations have occurred. The locks may be released in batches.

2.1.3 Locking in Parallel Sysplex

Lock Granularity

An important concept when designing a locking structure is the *granularity* of the database element to be serialized. Making this element very small (such as a record) reduces lock contention but increases the number of needed locks. Making the element bigger (like a set of tables) causes the opposite effect. There are options in IMS/DB and DB2 where the installation can define this granularity.

Hashing Technique

To avoid massive database lock structures in the CF, a *hashing technique* is used. Every element in the database to be locked has an identifier (ID). This ID is hashed, and the hash result is an index into the lock structure.

Synonyms

Because there are more database elements than lock entries, there is a probability of *synonyms*. A synonym refers to two or more distinct elements pointing to the same lock entry. When this situation happens, all the subsystems from different images that are using this lock structure need to perform lock resource negotiation.

Contention

During this operation, they can decide if the contention was *false* (just a synonym) or *real* (two subsystems trying to get access to the same lock).

Note that the lock requester may or may not be suspended until it is determined that there is no real lock contention on the resource. For example, he may be processing other work in parallel while the cross-system extended services (XES) makes the determination. Contention can be a performance problem for workloads that have a high level of sharing (locking) activity.

Real Contention: Real contention is a characteristic of the workload (set of locks that are obtained, lock hold time, and nature of workload concurrency), and is not easy to tune by tuning the Parallel Sysplex configuration. It is tuned by *tuning the workload itself,* for example, by avoiding long-running batch jobs concurrent with online, by avoiding running concurrent batch jobs that access the same data, increasing the frequency of checkpoints, and so forth.

Real contention is also reduced by reducing the degree of data sharing. CP/SM may for example help to make sure that transactions needing certain data are run on the same system.

False Contention: False contention, on the other hand, has to do with the hashing algorithm the subsystem uses and the size of the lock table in the lock structure. Assuming a reasonably efficient and "even" hashing algorithm, it is true that increasing the size of the lock table in the lock structure will reduce false contention. Carrying this out to extremes, by making the lock table arbitrarily large, you can make the false contention become as small as you like. This has practical limits, since, for IRLM, the only way to increase the size of the lock table is to increase the size of the lock structure as a whole, which may have sizing impacts. False contention should be kept low.

Recommendation for False Lock Contention

To minimize the effect of false contention, there are three options:

- Increase the lock structure size in the CF.
- Decrease the granularity of locking, or in other words, cut the number of locks. However, by decreasing the granularity of locking, you increase the probability of real contention at the expense of false contention.
- In the IRLM case, decrease the number of users connecting to the lock structure, thus allowing more lock entries within a given structure size.

How Much Contention is Acceptable?

Total contention is the sum of *real* and *false contention* and should be kept low. Recommendations show that the total locking contention should be kept to no more than 2.0%.

As guidelines, we recommend less than 0.1% false contention for CICS/DBCTL, the CICS/VSAM RLS and the GRS star workload and 1% for the IMS-TM/DB2.

Note: RMF reports total contention as well as false contention in the Postprocessor Coupling Facility Report (see 4.2.2, "RMF CF Activity - Structure Activity" on page 124).

2.1.4 Parallel Sysplex HW Components

Parallel Sysplex HW components include:

- CECs
- CFs
- Links

2.1.4.1 CECs Participating in Parallel Sysplex

IBM CECs participating in Parallel Sysplex

Table 3 explains how certain IBM CECs participate in a Parallel Sysplex environment.

Table 3. IBM CECs Participating in Parallel Sysplex

CEC	Support	Remark
IBM 9021 (711-based)	Can couple and be coupled to	ICMF, CFCC, and CF links
IBM 9121 (511-based)	Can couple	Cannot run CFCC using real CF links, but can use ICMF. Can have CF links to an external CF.
9672 (R1, R2, R3, G3*, G4*)	Can couple and be coupled to	ICMF, CFCC, and CF links
Multiprise 2000 (2003)**		ICMF, no links to external CF

Note: * G3 and G4 CECs may optionally contain ICFs

** IBM will make available in 1998 upgrades from the high end of Multiprise 2000 models to the 9672 processor family.

Some non-IBM CEC vendors have announced the ability to participate in a Parallel Sysplex environment. Contact these vendors directly if you have any questions on their products.

CF Hardware

A 9674 CF is a dedicated CEC which only runs the Coupling Facility Control Code. 9674 is based on same technology as the 9672.

2.1.4.2 The CF Function

XES SW Services

The Parallel Sysplex architecture has been implemented through a combination of functions:

- Cross System Extended Services (XES) services in OS/390
- A set of microcoded S/390 instructions (the CF function) that can operate in a logical partition

CFCC

The CF function is implemented in a logical partition running code called Coupling Facility Control Code (CFCC). The CFCC code runs:

- On a stand-alone 9674 (using external links)
- On any 9672-, 9021-711-based CEC (using external links)
- On any 9672-, 2003-, 9021-711-, 9121-511-based CEC (using (ICMF) emulated links)
- ICF on 9672- G3 and G4 based and followon CECs (using either emulated links (ICMF) or external links)

The CFCC code is identical in these environments. What makes each environment different is the resources it has access to (CP, CF links, and storage).

CFCC Code Levels

CFCC code has been enhanced several times. The original CFCC code (CFLEVEL 0) contained all the base functions needed to enable a Parallel Sysplex with the subsystem software levels of that time. Since then, CFLEVELs have delivered additional functional and performance enhancements. Cost evaluations may vary slightly depending on the CFCC code level. The following code levels are relevant at the time of writing:

- CFLEVEL 2 is the minimum level for CICS/VSAM RLS and provided performance enhancements for DB2 data sharing.
- CFLEVEL 3 is the minimum level for IMS shared message queues.
- CFLEVEL 4 adds functions that benefits IMS in particular, like dynamic change of structure size.

Dynamic CF dispatching capability

Enhanced dispatching algorithms enable you to define a backup CF in a logical partition on your system. This backup CF must have pre-allocated storage and it shares CPs with other active workloads. When idle it consumes less than 2% of a processor and when the backup CF is active, it uses only the CPU resource necessary to provide coupling (provided the partition has sufficient weight). Dynamic CF dispatching is supported on all coupling capable S/390 Enterprise Servers. Prior to this capability, a CF with shared engines needed to be used with great care and needed to be given significant resource due to the “active wait” algorithm of its dispatcher.

Internal Coupling Facility (ICF)

For all S/390 G3 and G4 Enterprise Servers, except 10-way systems, the IBM S/390 Internal Coupling Facility (ICF) allows a dedicated spare processing unit to function as a CF. One or two ICF features can be ordered utilizing spare processing units as a CF with either emulated (ICMF) or external CF links. For production environments, the ICF running CFCC can connect directly to Parallel Sysplex-enabled production systems using CF links. This option can provide a lower cost alternative for a back-up CF to a 9674. It could also provide a low-cost CF for a single CEC installation.

ICMF

The Integrated Coupling Migration Facility (ICMF) is an alternative CF function. It is primarily used in a test or a migration configuration because it emulates the coupling links. *ICMF cannot run on the IBM 9674 CF*.

Participating OS/390 images must run in a logical partition in the same CEC as the ICMF logical partition. One OS/390 image cannot couple to ICMF and CFCC at the same time. Up two ICMF logical partitions can run in one CEC.

For a discussion on how to size the logical partition running either CFCC (also referred to as a CF partition) or ICMF (also referred to as CF/I partition) refer to A.2, “LPAR/Capacity Estimator (LPAR/CE)” on page 253.

CF Structures

All the images in the Parallel Sysplex configuration may use *structures* to communicate. The *structures* reside in the storage of the CF partition.

Storage in the CF is used for several purposes:

- By the CF itself (HSA, CFCC, DUMP)
- For communication (list structure)
- For locking (lock structure)

- For database data (cache structure)
- For redundancy (backup)

CF storage can be manipulated by applications via services provided by OS/390 and the CF.

Before an application can use a CF, information necessary to define the CF structures must be provided in the CFRM policy. Table 4 summarizes the IBM products and functions using the CF. For each structure, you need to make estimates based on your specifications. It is important to ensure performance and availability characteristics, where the structures will reside. An overview of all structures and a discussion on structure sizing, placement and rebuild characteristics is provided in *OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook*, SG24-2076. You may adjust the structure size and placement based on application usage, performance and CFLEVEL.

Table 4 (Page 1 of 2). CF Structures for IBM Products

Component	Function	Structure Type	Where Documented
XCF	Signalling	List	*OS/390 MVS Setting Up a Sysplex*, GC28-1779
JES2	Checkpoint Data Set	List	*OS/390 MVS Initialization and Tuning Guide*, SC28-1751
Allocation	Tape sharing	List	*OS/390 MVS Initialization and Tuning Guide*, SC28-1751 *OS/390 MVS System Commands*, GC28-1781
VTAM	• GR • MNPS	List	*VTAM V4R4 Network Implementation Guide*, SC28-1913
RACF	RACF Database	Cache	*OS/390 MVS Security Server System Programmer's Guide*, SC28-1913
IRLM	Locking	Lock	*OS/390 Sysplex Application Migration*, GC28-1863 *IMS/ESA V6 Administration Guide: Database Manager*, SC26-8725 *DB2 for OS/390 V5 Data Sharing: Planning and Administration* SC26-8961
IMS	• OSAM • VSAM • VSO DEDB • SMQ • EMH	• Cache • Cache • Cache • List • List	*OS/390 Sysplex Application Migration*, GC28-1863
DB2	• SCA • GBP	• List • Cache	*DB2 for OS/390 V5 Data Sharing: Planning and Administration*, SC26-8961
CICS	Temporary Storage sharing	List	*CICS TS for OS/390 System Definition Guide*, SC33-1682
DFSMS VSAM/RLS	VSAM sharing CICS	• Lock • Cache	*DFSMS/MVS V1.3 Presentation Guide*, GG24-4391

Table 4 (Page 2 of 2). CF Structures for IBM Products			
Component	**Function**	**Structure Type**	**Where Documented**
System Logger	Logging for: • Syslog • Logrec • CICS/ESA • IMS/ESA V6	List	*OS/390 MVS Programming: Assembler Services Guide*, GC28-1762 *OS/390 MVS Programming: Assembler Services Reference*, GC28-1910
GRS Star	Resource Serialization	Lock	*OS/390 MVS Planning: Global Resource Serialization*, GC28-1759
SmartBatch	Parallel Batch	List	*IBM SmartBatch for OS/390*, GC28-1627

Define CF Storage as Central Storage

CF storage can be defined as central storage (CSTOR) and expanded storage (ESTOR). Certain parts of the CFCC code and the structures must be in CSTOR. Other parts may be in ESTOR.

Always use CSTOR for the CF if possible. Structure space planning is complicated if you use both types of storage. ESTOR should only be used if you require a CF partition with more than 2GB of storage. You should first define all storage up to 2GB as CSTOR.

Availability Considerations

For availability, you should plan for several (n+1) CFs. In case one fails or need to be taken out for service, you can rebuild the structures from the failing CF, and you need spare capacity in the other CFs to allow that to happen. The capacity required depends on the number of CFs, and whether all structures need to be rebuilt.

Hot Standby CFs

The idea of having a ″hot standby″ CF running on one of the CECs in a production sysplex is to reduce cost compared to buying a second IBM 9674. The hot standby CF should have few or no structures defined in order to rebuild the structures if the stand-alone CF fails. It is highly recommended to use the dynamic CF dispatching capability for hot standby CFs. This dramatically reduces the need for CF resources when the hot standby is idle. A more detailed discussion on hot standby CFs is provided in *OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook*, SG24-2076.

One-way versus N-way CFs and CF utilization

Up to 40% average CPU utilization is seen as an acceptable upper utilization limit for which acceptable response times for a CF request are maintained. Actually you may run at higher utilizations; the rationale for the recommendation is to allow for one (or n-1) CF to absorb all (n) CF work in case of CF or CF connectivity failure. Usually N=2 at present, and so we are saying that in the case of a CF failure we will load the surviving CF to 80%.

Keeping the 40% in mind, it is of almost no importance whether CF requests are handled by a CF with multiple small CPs, or by a CF with one large CP, given the same total processing power. Of course, a large CP will handle requests faster than a small CP, given the same queue length.

For a detailed discussion about different CF configuration options, refer to Chapter 5, “A CF Processing Model” on page 143.

CF CPU Speed

You should consider the CPU speed of the CF relative to the CPU speed of the sending CEC. Table 8 on page 66 gives an overview of the enablement costs for DB2, IMS DB, and CICS/VSAM RLS data sharing. The data sharing overhead is dependent from the speed of both the CEC and the CF.

Changing from a slower CF model to a faster CF model may result in performance improvements for request response times. IBM laboratory measurements for different CFs (none of which were using HiPerLinks) show the following:

- Changing from an IBM 9674-C01 CF to an IBM 9674-C02 CF may decrease the synchronous request response time by 15 to 20 microseconds. The lower value of the spread is for lock requests and the higher value is for cache requests.
- Changing from an IBM 9674-C02 to an IBM 9674-C04 decreases the synchronous request response time by 25 to 45 microseconds. The lower value of the spread is for lock requests on an IBM 9674-C04 with up to 5 CPs. The higher value is for a cache request on an IBM 9674-C04 CF with 6 to 10 CPs.
- Changing from an IBM 9674-C02 to an IBM 9674-C05 decreases the synchronous request response time by 40 to 70 microseconds. The lower value of the spread is for lock requests and the higher value is for cache requests.

You may observe slightly different results in your environment.

CF Links

The CFCC uses *coupling links* to connect coupling capable CECs with the CF partition. Two types of links currently exist:

- The single mode fiber (9/125 micron fibers) allows a distance of 3 Km and a speed of 100 MB/sec. Any distances above 3 Km between a CEC and a CF require an Request For Price Quotation (RPQ).
- The multimode fiber (50 micron fibers) is supported with a distance of 1 Km at a speed of 50 MB/sec.

Always use single mode fiber if possible. Refer to Table 5 on page 60 for the impact of using multimode fibers versus single mode fiber. Notice that both types of fibers can use HiPerLinks. The table applies to DB2 and CICS/VSAM RLS environments. The effect is smaller in IMS V5 data sharing environments.

Table 5. Multimode CF Link Effect on ITR					
	CF				
Sending CEC	**ITR Degradation**				
	9672-E/P/R1 9674-C01	**9672-R2/R3 9674-C02/C03**	**9672 G3 9674-C04**	**9672 G4 9674-C05**	**9021 711**
9672-E/P/R1	0%	0%	0%	0%	0%
9672-R2/R3	0%	0-1%	0-2%	0-3%	0-3%
9672 G3	0%	0-1%	0-2%	0-3%	0-3%
9672 G4	0%	0-1%	0-2%	0-3%	0-3%
9021 711	0%	0-1%	0-2%	0-3%	0-3%
Note: Applies to 100% DB2 or CICS/VSAM RLS data sharing environment.					

For a discussion on the effects of coupling link *distances*, refer to 6.2.1, “Defining Your Environment to Quick-Sizer” on page 165.

High Performance Links (HiPerLinks)

The IBM 9672 G4 models are equipped with HiPerLinks. The IBM 9672 G3 models can be upgraded to use HiPerLinks. HiPerLinks are a faster CF link technology that mainly improves the performance on the sender side. The performance characteristics of HiPerLinks show response time improvements ranging from 15% to 40% for asynchronous requests and up to 25% for synchronous CF requests. The larger the data transfer, the greater the improvement. On average, the link capacity may be improved 20%, and data sharing overhead may be reduced by 10%.

HiPerLinks can connect to HiPerLinks or older CF link technology. They cannot be combined with older CF links on the same CEC

The gains you achieve by upgrading to HiPerLinks in response time are close to the response time gain that is achieved by replacing an IBM 9674-C02 with an IBM 9674-C04 CF. HiPerLinks also make it more feasible to let XCF use CF structures instead of CTC connections for signalling.

The following XCF exploiters are likely to benefit from HiPerLinks:

- GRS (ring)
- CICS MRO
- SMSQ (especially for large messages)
- BatchPipes
- DB2 (especially when using 32K buffers)
- VSAM RLS (with large CI sizes supported)
- CICS temporary store support
- VTAM high performance routing (HPR)

Effect of Distance on CF Links

Link distance (the distance between CEC and CF) partially determines the performance of the individual CECs in a Parallel Sysplex. Generally, the hardware component of time spent in the CF (the CF service time as it shows up in RMF) increases 10 microseconds for each kilometer added to the length of the connecting fiber.

Synchronous accesses take more time as the CF link increases in length. This translates to a reduction in ITR in the sending CEC.

OS/390 tries to optimize the use of synchronous and asynchronous requests and their usage is software level-dependent. APAR OW15588 describes a small programming enhancement (SPE) which makes changes in this area.

Table 6. Reduction in Sender ITR/km CF Link (Because of Synchronous CF Access)

Sending CEC	
9672-E/P/R1	.2%
9672-R2/R3	.2-.3%
9672 G3	.2-.4%
9672 G4	.7-.9%
9021 711	.7-.9%
Note: These are IBM Benchmark figures.	

Thus we see that, for example, a 10 km CF link causes a 7-9% ITR reduction on an IBM 9021-711 based CEC connected to an IBM 9674-C02 for the standard IBM benchmark workloads. These overheads are heavier than most real-life systems and many systems would incur less than half that cost.

Link utilization will also increase with distance by 1-2% per kilometer, depending on load.

Capacity Planning Considerations

The number and type of CFs must be decided. *OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook*, SG24-2076 contains information and examples about how to estimate the following resources:

- CP resources in the CF to run the CFCC or the ICMF code. The CP usage depends on the number and the definition of the structures, as well as the workload.
- Storage in the CF into which the structures are put. The storage size depends on the number of structures, the subsystems used in the OS/390 images, the characteristics of workloads used in data sharing mode, and the CF parameters specifications. The processor storage for the CF should be defined as Central Storage (not as Expanded Storage). The total storage in the CF includes storage for CFCC (or ICMF) itself, HSA, dumpspace, and “spare” storage, to allow structure rebuild from other CFs. OS/390 allows dynamic structure size changes. Certain subsystems can continue when structures need to be resized. The size can be changed up to a maximum size determined at allocation time.
- The number, type and usage of coupling links

Modelling CF Resources

All the CF resources mentioned above are modelled in more or less detail by CP2000 and Quick-Sizer. Refer to Chapter 6, “Using CP2000 for Capacity Planning in a Parallel Sysplex” on page 161 for further information.

For a discussion on how to size structures for data in DB2, IMS/DB, and CICS VSAM/RLS, refer to:

- 2.2.5, “DB2 Buffer Pool Sizing in Parallel Sysplex” on page 70.

 DB2 uses a *store-in* technique to store its group buffer pool in a CF cache structure.
- 2.3.5, “IMS/DB Buffer Pools Sizing in Parallel Sysplex” on page 80.

IMS/DB uses the *store-through* technique for OSAM structures and *directory-only* technique for VSAM structures (IMS V6). For IMS/V6 VSO DEDB block level sharing uses a *store-in* buffer pool.

- 2.5, “CICS/VSAM RLS Data Sharing Considerations” on page 84

 CICS VSAM/RLS uses the *store-through* technique for its CF structures.

Note: Reporting tools, such as RMF, monitor CF resources and can provide you detailed information.

2.1.5 Cost Factors in Parallel Sysplex

Locking and caching activities are normally responsible for most of the CF activity. Other items are usually less significant. Table 7 describes the relative importance of certain functions with respect to increase in (CEC) resource requirement. Similar weighting factors are currently used by CP2000 and Quick-Sizer as defaults when doing CF access modelling. They can be overruled by the user.

Relative Parallel Sysplex Cost Factors

Table 7. Relative Importance to Increase in Resource Requirement

FUNCTION	WEIGHT
Update serialization (locking)	5 to 10
Local Buffer Coherency	3 to 10
Shared Message Queue	7 to 10
VTAM+JES2+RACF	0 to 2
XCF CICS-MRO	1 to 5
Other XCF messaging	0 to 1
DASD I/O Increase	0 to 5

The numbers in the column on the right are not percentages; they are relative contributions to resource usage.

- **Update Serialization** is locking. This is often the largest contributor to resource usage in the IMS/DB data sharing environment. We will discuss in some detail later how you can estimate the number of locks in your environment.

 For a detailed discussion, refer to 2.2.4, “DB2 Locking Rate Calculations” on page 67 or 2.3.4, “IMS/DB Locking Rate Calculations” on page 78.

- **Local Buffer Coherency** is related to cache accesses and buffer invalidations.

 This is often the largest contributor to resource usage in the DB2 data sharing environment.

- **IMS Shared Message Queue**

 This is a major contributor to resource usage at about the same level as a DB manager.

- **VTAM, JES2 AND RACF** CF access frequency is usually low compared to locking rates and resource usage is correspondingly less.

- **CICS Transaction Routing** cost is small to fair and is based on the number of transactions routed. Transactions are routed to balance the load of processing on each system in the Parallel Sysplex. The CECs in a Parallel Sysplex are generally balanced without routing more than 5% to 10% of the transactions. If your system is set up in such a way that a few TORs are

routing to a variety of AORs, the overhead is higher, probably not to exceed 3% or 4% when routing all transactions. Cloned systems only increase about 1%.

- **Other XCF Messaging** contributes only a small overhead.
- **DASD I/O Increase** is caused by two types of processing.
 - When buffers are invalidated due to updates, the next reference to the data causes an I/O or an access to the CF, depending on the subsystem, (that would have formerly been a buffer hit).
 - When multiple transactions are running on a single system using the same data, subsequent transactions can reuse data in the buffer without re-reading. After migration to Parallel Sysplex, these same transactions may run on different systems. On a different system, the data has to be re-read; this increases I/O. From buffer traces, the increase in I/O per transaction is about 1% to 5% (average about 3%) for each IMS/DB system added to a Parallel Sysplex. For DB2 the extra number of *accesses* to the CF is very little and close to 0%.

The buffer invalidation effect on number of I/Os are discussed further in:

- 2.2.2, “No Expected I/O Growth for DB2” on page 64.
- 2.3.2, “Expected I/O Growth for IMS/DB” on page 76.

2.2 DB2 Parallel Sysplex Data Sharing Considerations

This section describes some of the DB2 capacity planning considerations that are important when migrating from a single OS/390 image system to a Parallel Sysplex.

We discuss the following topics:

- Expected I/O growth, where the majority of these are accesses to CFs
- Parallel Sysplex CPU cost
- Locking rate calculation
- Buffer pool sizing

Three key DB2 Data Sharing Publications

For a discussion on performance and capacity planning topics related to DB2 Parallel Sysplex data sharing, refer to:

- *DB2 for OS/390 V5 Data Sharing: Planning and Administration*, SC26-8961
- *DB2 for MVS/ESA V4 Data Sharing: Performance Topics*, SG24-4611
- *S/390 MVS Parallel Sysplex Performance*, SG24-4356

Another redbook, which discusses the DB2 Capacity Planning, mostly related to single system environments, is *DB2 for OS/390 Capacity Planning*, SG24-2244.

DB2 CP2000 Case Study

For an example on how CP2000 support DB2 Parallel Sysplex capacity planning, refer to the step-by-step explanation provided in 7.1, “Case Study: DB2 Migration to a Data Sharing Environment” on page 197.

2.2.1 DB2 Parallel Sysplex Data Sharing

DB2 V4 can improve data availability and increase overall system capacity with its implementation of data sharing across multiple CECs, exploiting Parallel Sysplex technology.

DB2 Data Sharing Group

DB2 data sharing allows up to 32 DB2 subsystems, joined by one or more CFs, to have concurrent read and write access to the same set of data. These DB2 subsystems, known as a *data sharing group,* use a single catalog and directory which provides a single definition for all DB2 objects.

DB2 data sharing can improve capacity, transaction throughput, and the overall cost of computing. DB2 data sharing provides data integrity, flexibility for growth, and workload balancing. After you have defined one or more data sharing groups and its members, data can be accessed by any DB2 in the group. Before you start DB2 for data sharing, you define one IRLM lock structure, one list structure, and at least one cache structure.

DB2 CF Structure Usage

Lock structure There is one lock structure per data sharing group used by IRLM to control locking.

List structure There is one list structure per data sharing group used as the shared communication area (SCA). The SCA contains, for example, database exception information.

Cache structures These structures are used as group buffer pools (GBP) for the DB2 data sharing group. These GBPs are used to cache data that is of interest to more than one DB2 and to maintain data consistency across buffer pools by using a cross-invalidate mechanism.

DB2 Availability Recommendation

We strongly recommend that you plan for and define the availability characteristic of the SCA, the lock structure and the cache structure for loss of connectivity failures. For additional information, refer to *DB2 for OS/390 V5 Data Sharing: Planning and Administration*, SC26-8961.

2.2.2 No Expected I/O Growth for DB2

This discussion assumes the local buffer pools have the same size as that for non-data sharing and that the database sizes are unchanged after the migration to DB2 data sharing. This is further discussed in 6.4.3, “I/O Projections” on page 194.

GBPCACHE ALL? or GBPCACHE CHANGED?

There are two options when you use database buffers in the CF. In *addition* to having them in your local OS/390 systems, you may choose whether the CF is caching *updated* data or it is caching *all* data. These options apply to both DB2 data and index pages. These are specified on a *buffer pool level*.

No I/O Growth

If you cache data in the CF and there is enough space in the CF for caching (no reclaims occur), then there are assumed to be no changes in the number of DB2 related DASD I/Os. By caching all data in the CF, there will be reads from the CF, that would not occur if you only had a buffer pool in a single system.

CP2000 uses a default value of 0% increase. This is further discussed in Chapter 6, “Using CP2000 for Capacity Planning in a Parallel Sysplex” on page 161. If you cache only updated data in the CF and if there is enough space in the CF for caching these pages, then there is only a very slight increase in the number of DB2 I/Os (some *read only* pages may now be read several times). Since DB2 now places the changed data in the group buffer pool (GBP), most of the increases in local buffer pool cross-invalidation misses are satisfied by retrieving the data from the global buffer pool in the CF. The increase in DB2 related I/Os is assumed to be about 0% per added OS/390 image. Note that DASD I/Os are to an extent replaced with accesses to the CF.

2.2.3 Parallel Sysplex CPU Cost for DB2 Data Sharing

While the CF buffer is being updated and until the buffer invalidation has completed in the other CECs hosting DB2s in the data sharing group, the requesting CP remains busy. Using the CF causes processor cycles at the requestor CECs to be “wasted.” This work is charged to the cross Coupling Facility (XCF and XES) function of the requestor. The faster the speed of the CF (everything else being equal) the lower the overhead. If the CP of the requestor CEC is faster than that of the CF, the overhead is higher.

Experience-based numbers are available, and CP2000 bases many of its algorithms on these numbers. They depend on:

Factors Influencing DB2 Data-Sharing Cost

- The number of CECs
- The requestor CEC CP speed
- The CF CP speed
- The Coupling Facility Control Code (CFCC) level
- The SCP and subsystem release
- The number of DB2s in the data sharing group
- The link speed and distance
- The degree of data sharing
- The R/W ratio
- The data access pattern
- The type of DB2 indexes used
- DB2 bind parameters (affecting locking rates)
- How often applications commit their data

CP2000 provides you the opportunity to examine the needed capacity of a Parallel Sysplex exploiting DB2 data sharing. The following formula is used as a rule of thumb.

Note: This formula is identical to the one used for IMS/DB data sharing as discussed in 2.3, “IMS/DB Parallel Sysplex Data Sharing Considerations” on page 74.

DB2 Parallel Sysplex Data Sharing Formula

DB2 Parallel Sysplex Data-Sharing Cost Rule Of Thumb:

$$A = M\% \times T + (E\% + ((N - 2) \times I\%)) \times D$$

where:

A = Additional RPP needed to support multisystems management and data sharing in a Parallel Sysplex
T = Total RPP of the OS/390 system you are sizing
D = RPP of the application that is sharing data

D includes the following elements:

- RPP for DB2, including its use of OS/390 and VTAM services
- RPP for CICS or IMS/DS, including its use of OS/390 and VTAM service

N = Number of OS/390 images participating in the Parallel Sysplex
M = Multisystems Management cost associated with a Parallel Sysplex (3%)
E = Enablement cost for data sharing in a Parallel Sysplex
I = Incremental growth cost for data sharing in a Parallel Sysplex

For a discussion of the term Relative Processing Power (RPP), refer to 1.3.1, "From Cycle Time to M-value" on page 8.

Table 8 shows the coefficient E (from above formula). Notice that the enablement costs for DB2 and IMS DB data sharing are about the same. Customer experience is often **less then half** these figures, so that the table represents a worst case scenario.

Table 8. Coefficient (E) for Enablement Cost of IMS DB, DB2 and CICS/VSAM RLS Data Sharing

	Coupling Facility				
Sender CEC					
	9672-E/P/R1 9674-C01	**9672-R2/R3 9674-C02/C03**	**9672-G3 9674-C04**	**9672-G4 9674-C05**	**9021 711**
9672-E/P/R1	14	13	12	12	12
9672-R2/R3	17	15	14	14	14
9672 G3*	21	18	16	15	15
9672 G4	21	18	16	15	15
9021 711	23	20	19	17	17

Note: 100% data sharing environment at CFLEVEL 2 minimum; all values are percentages; engine speed for G3/G4 CECs and corresponding C04/C05 CFs vary slightly depending on the model; numbers are based on IBM benchmarks; most customer experiences indicate **less than half** the numbers shown in this table.

For IMS: RLKC = 0.25, **For DB2:** RLKC = 0.33,

For G3*: 9672 G3 with HiPerLinks will show lower enablement cost of about 1-2%.

The incremental growth coefficient (I) is approximately 0.5% for all CEC models. In other words, the scalability in Parallel Sysplex is nearly linear.

Note:

Total Cost of Data Sharing is Usually Less Than 10%

- These numbers are approximations based on the IBM benchmarks doing 100% data sharing. In average scenarios, the *total* cost required to support data sharing in Parallel Sysplex is usually less than 10%. "Total" here includes non data sharing workloads as well.
- Because of the small incremental growth percentage, 32 images using 9672 CECs provide the largest single system image in today's online transaction processing marketplace.
- CP2000 does not directly use this formula; however its calculations are based on similar concepts.

Initial Sysplex Cost

There is an *initial cost* to run in a base sysplex or a Parallel Sysplex. This is sometimes referred to as the multisystems management cost (M), and is approximately 3% of the total workload. M includes such functions as GRS ring-processing, JES2 shared checkpoint-processing and workload management. *If you are already using GRS or are running in a sysplex, you already paid this cost or a large portion of it.* The initial cost applies to all systems participating in the Parallel Sysplex.

The data-sharing cost then consists of two components: the enablement cost (E) and the incremental growth cost (I). Both of these costs depend on the technology used. The enablement cost is only related to the data sharing part of the workload. The incremental growth cost does *not* apply unless you have more than two systems.

2.2.4 DB2 Locking Rate Calculations

DB2 will issue a number of accesses to the CF in a Parallel Sysplex data sharing environment. These accesses include:

1. LOCK/UNLOCK requests
2. Physical reads (directory)
3. Updates (cache)
4. Reads of "buffer invalidated" data

In the following topic we focus on the LOCK/UNLOCK requests.

DB2 Locking Rates are Key to Data Sharing Capacity Planning

Locking rates are key to data sharing performance. Table 7 on page 62 provides a high level view of contributors to data sharing overhead.

This section provides a short discussion of locking rates for DB2, depending on the DB2 versions.

When you plan to migrate from single DB2 systems to a Parallel Sysplex using DB2 data sharing, it is important to estimate your current locking rate. Today no report shows these locking numbers explicitly; they must be calculated. In the following topic, we discuss how to calculate locking numbers and where to find the data.

Locking Rates and SMF Records

The input data for the DB2 locking rate calculation is based on SMF records type 100 and 101 written by DB2 and known as statistic and accounting records.

After running the DB2 Performance Monitor (DB2PM), the created report provides us with the following numbers useful for calculation of the locking rate:

LOCK request numbers
UNLOCK request numbers
CHANGE request numbers

7.1, “Case Study: DB2 Migration to a Data Sharing Environment” on page 197 shows an example on how to extract the LOCK, UNLOCK and CHANGE request numbers from DB2PM reports. The *total* locking rate is the sum of the various types of requests:

$$Locking\ rate = \frac{(LOCK\ requests + UNLOCK\ requests + CHANGE\ requests)}{Time}$$

DB2 Locking Rate Is a Function Of the DB2 Release

The total locking rate is sensitive to the *DB2 version* and the *type of indexes* used. When migrating between various DB2 versions or migrating to Type 2 indexes, the locking rate changes as described in the following:

$$Locking\ rate = \frac{((LOCK\ requests + UNLOCK\ requests + CHANGE\ request) \times m)}{Time}$$

where *m* is an experience based migration factor.

DB2 Locking Rate Reduction:

Several different DB2 migration scenarios are listed below:

V2 => V3

Use m=0.67

V3 => V4 Index Type 1

From DB2 V3, to DB2 V4 using Index Type 1, the locking rate remains almost unchanged.

Index Type 1 => Index Type 2

Use m=0.5 as a first pass but you can choose other values based on the following discussion.

Importance of R/W ratio

Locking reduction when moving from type 1 to type 2 indexes is expected to vary between 25% and 75%, where 25% relates to environments with a low R/W ratio and 75% relates to an environment with a high R/W ratio. IBM laboratory measurements show 50% savings with a R/W ratio of 4. The 25% reduction in locking rate reflects a R/W ratio of 1, and the 75% reduction in locking rate reflects a R/W ratio of 8.

The R/W ratio is defined as:

$$R/W\ ratio = \frac{\#\ SQL\ Reads}{\#\ SQL\ Writes} = \frac{\#\ Selects + \#\ Fetches}{\#\ Inserts + \#\ Updates + \#\ Deletes}$$

This information is readily available from DB2 statistics.

Also check the redbook *DB2 for OS/390 V5 Performance Topics*, SG24-2213 which contains a chapter about reducing data-sharing costs.

V2 => V4 or V2 => V5

If you want to estimate the lock reductions when you migrate from DB2 V2 to DB2 V4 Index Type 2, you have to apply two factors from the previous rules. In this case you multiply by *0.67 and by 0.5*.

Summary of suggested m values

1. m ≈ 0.67, if migrating from DB2 V2 to V3 or V4 or V5 with Index Type 1
2. **m ≈ 0.34, if migrating from DB2 V2 to V4 or V5 with Index Type 2**
3. m ≈ 1.0, if migrating from DB2 V3 to V4 or V5 with Index Type 1
4. **m ≈ 0.5, if migrating from DB2 V3 to V4 or V4 with Index Type 2**
5. **m ≈ 0.5, if migrating from V4 or V5 Index Type 1 to Index Type 2**
6. **m ≈ 0.9, if migrating from V4 Index Type 2 to V5 Index Type 2**[5]

Default CP2000 RLKC

If you have not gone through the calculations above, you can let Quick-Sizer provide you with a default lock access rate. Quick-Sizer uses the following formula:

$$DB2_{\text{lock access rate}} = 0.33 \times \text{M-value}$$

The factor 0.33 is an assumption and relates to experience for DB2 V4, Index Type 2. The typical range for RLKC is 0.25-0.50. This number fluctuates among installations. If you think that the factor 0.33 does not reflect your system, you may change the lock access rate numbers. For a short discussion of the this term, refer to 1.4.3, “Relative Lock Content (RLKC)” on page 34. In Chapter 6, “Using CP2000 for Capacity Planning in a Parallel Sysplex” on page 161 you will see how Quick-Sizer can be used to model the migration effects when moving to a Parallel Sysplex.

2.2.4.1 DB2 V5 Locking Considerations

In DB2 V5, Sysplex Query Parallelism support was included. When exploited, this function will be using the same DB2 structures for locking, cashing, and recovery as others. However, the application of the structures will be different from for example, OLTP. The Sysplex Query Parallelism support will mostly be read-only access, with possibly a lot of Uncommitted Reads. This will imply limited coupling overhead. If the Query Parallelism function is sharing data with OLTP, it will have to register interest, but still little overhead compared to normal OLTP work.

Also, DB2 V5 has Partition Level Locking to further isolate Read/Write activity from a DB2 to one partition from another partition. This function is implemented in order to avoid inter-DB2 Read/Write interest.

[5] This is the average lock rate reduction due to the selective partition locking enhancement introduced in DB2 V5. Note, many customers will see only a little improvement whereas a few will experience a big lock reduction.

2.2.5 DB2 Buffer Pool Sizing in Parallel Sysplex

When data sharing is active, DB2 has local buffer pools in the OS/390 systems as well as globally in the CF. Figure 14 illustrates local buffer pools (BP) and global group buffer pools (GBP). Note that DB2 local buffer pools *include both virtual buffer pools and hiperpools.*

On the left side of the figure you see a single OS/390 system. In this system all DB2 buffer pools are local to the system (OS/390 1) The buffer pools and databases are, in other words, not shared between DB2s in one or more OS/390 systems. On the right you see a Parallel Sysplex with n copies of OS/390 systems and a CF. Here, local DB2 buffer pools are shown in the OS/390 systems (BP) and group buffer pools are shown in the CF. (GBP). The DB2 database is shared with OS/390 systems that belong to the data sharing group. Data can be accessed immediately either from the local buffer pools (virtual buffer pools or hiperpools) or from the group buffer pool. Data is written in a fashion similar to that of “deferred write” in a single system.

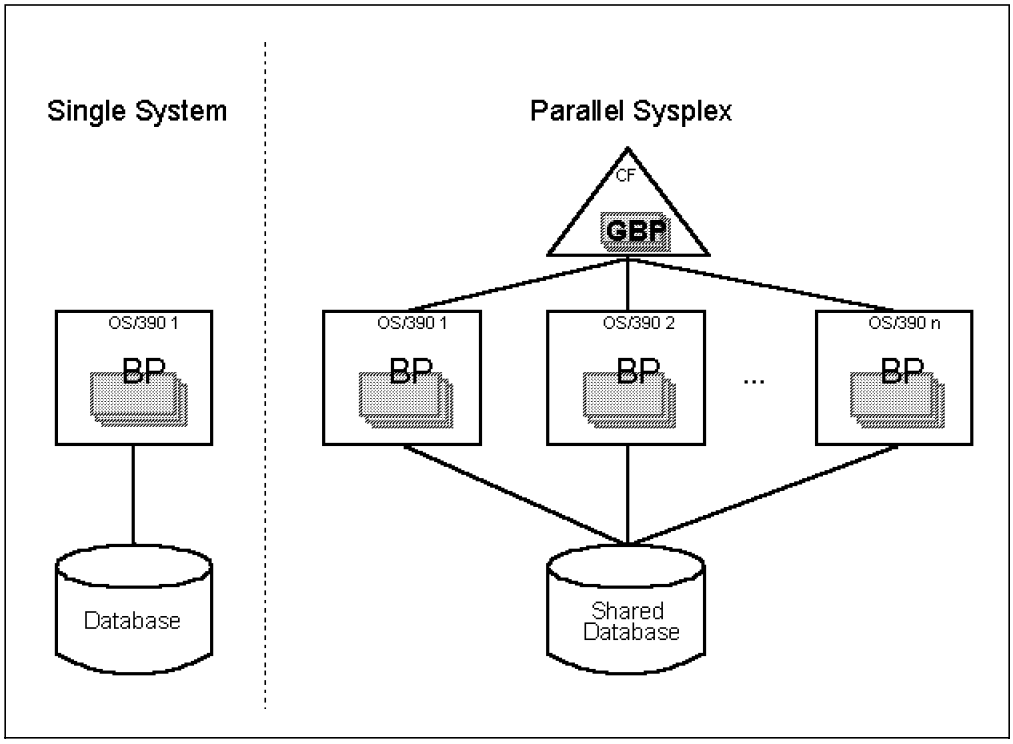


Figure 14. DB2 Buffer Pools, Single OS/390 versus Parallel Sysplex

Factors Influencing DB2 Buffer Pool Sizing

In the following are listed several factors that affect both local buffer pool and group buffer pool sizing for DB2 in a Parallel Sysplex.

- The percentage of data sharing
- The access pattern of the transactions and whether access patterns are different among the OS/390 systems
- The update frequency of transactions
- The selected CF GBP caching technique, caching all or caching updated buffers only. The caching all technique is not currently recommended for performance reasons.
- The castout thresholds for freeing buffer space in local buffer pools (BP) and in global buffer pools (GBP)
- The number of DB2s in the data sharing group

For simplicity we will distinguish between the following DB2 data sharing scenarios:

- Caching only changed data in the CF (GBPCACHE CHANGED)
- Caching all data in the CF (GBPCACHE ALL)

Let us discuss the different scenarios to see how various factors affect buffer pool sizing.

CP2000 calculates the default group buffer pool size as 50% the sum of all the local buffers.

2.2.5.1 Caching Only Changed DB2 Data in the CF

This option is specified in the clause GBPCACHE(CHANGED) to the DB2 ALTER TABLESPACE command.

Note: Updated pages are written to the group buffer pool *only* when there is inter-DB2 interest.

Similar Access Pattern

If all DB2 systems are cloned to be identical to each other, then the access pattern for DB2s in the data sharing group is very similar. In the following we discuss how to size the local and global buffer pools in an environment with similar access pattern. Local buffer pools should be read as including hiperpools.

GBP and BP Sizing Rules Of Thumb

Local buffer pools in each OS/390 should have the *same* size as in the single OS/390 case. The size of the group buffer pool (GBP) depends on the change activity and the access pattern of data, and varies *from 10% to 40% of the sum of the local buffer pool sizes* of the OS/390 systems. Include hiperpools in the sum of the local buffer pool sizes.

Lock Structure Sizing

For the global IRLM lock table in the CF you should have about 55% of the sum of the local lock tables.

SCA Structure Sizing

For the SCA (communication area) in the CF, you should have something like 10 MB to 50 MB, depending on the size of the data sharing system.

An Example If you have a 400 MB buffer pool in the OS/390 system, then in the Parallel Sysplex each OS/390 system also has 400 MB buffer pool; smaller buffer pools per OS/390 are *not recommended.* This is primarily because the access pattern of the transactions are similar to the single system scenario, even though you have a lower transaction rate per system.

Since the access pattern is similar in the systems, the local buffer pools will contain approximately the same data. For this reason the logical sum of all the local buffer pools is 400 MB. Your group buffer pool thus needs from 20 to 160 MB (400 MB times 10% to 40%), say 160 MB. Choose 30 MB for the SCA. Assume that 10 MB is required for the lock structure. In total DB2 data sharing would require allocation of 160 + 10 + 30 MB of CF storage.

For more information on these calculations, refer to *DB2 for OS/390 V5 Data Sharing: Planning and Administration*, SC26-8961.

Different Access Pattern

In the case of different access patterns, buffer pools sizes in the local OS/390 systems should be somewhere between:

- The previous single OS/390 system buffer pool size divided by the number of systems n
- The previous single OS/390 system buffer pool size

GBP and BP Sizing Rules Of Thumb

For the group buffer pool you will need from 10% to 50% of the logical sum of the local buffer pools. However it is probably less (closer to 10%) than the case where the access patterns were the same.

For more information on these calculations, refer to *DB2 for OS/390 V5 Data Sharing: Planning and Administration*, SC26-8961.

2.2.5.2 Caching All DB2 Data in the CF

This option is specified in the clause GBPCACHE(ALL) to the DB2 ALTER TABLESPACE command.

Note: Currently this method is not generally recommended as it requires large CF buffers, and it increases the traffic to the CF. However this option may be used to *lower* the storage requirements in the local buffer pools.

By specifying this option you ensure caching of all DB2 data in the group buffer pool. This option can be specified per DB2 tablespace and works for both data and index pages.

GBP and BP Sizing Rules Of Thumb

Similar Access Pattern

Different Access Pattern

Let us first look at the case where *the access pattern is similar* in each local OS/390 system. In this case, the most recently used buffers in each of the local systems tend to be identical. The size of the local virtual buffer pools are also equal to the size of the single system. The same set of buffers are now placed in the CF as well, since you specified the option to cache *all* buffers. The size and content is very similar to the buffer pools in the local OS/390 system. Let us discuss what happens in the case where *the access patterns are different* among the local OS/390 systems. In this case the required sizes of the individual local virtual buffer pools are smaller than the single system virtual buffer pool. How much smaller, you might ask? Again, it depends on the many factors described earlier. The buffers in the local OS/390 systems will all be cached in the CF, too.

The size of the group buffer pool will be the *sum of the local virtual buffer pools*. It is the *logical sum*, since no data element has more than one copy in the CF.

Note: The sum does not include hiperpools since group buffer pools and hiperpools are mutually exclusive. The size of the group buffer pool will therefore be somewhere between:

- The previous single system buffer pool size
- The arithmetic sum of all the local buffer pools

The size of the group buffer pool depends on the change activity and the access pattern of data and to what extent GBPCACHE(ALL) has been specified. It is assumed to vary *from 50% to 100% of the sum of the local buffer pool sizes* of the OS/390 systems.

Independent of the access patterns, the CF should always have enough storage for the global IRLM lock table and for the communication area (SCA).

Refer back to 2.2.5.1, “Caching Only Changed DB2 Data in the CF” on page 71 for a short discussion on how to size the lock table and SCA.

Let us look at an example. If you have a 400 MB buffer pool in the single OS/390 system case, what will you need in the Parallel Sysplex? In the Parallel Sysplex in each local OS/390 system you now have:

- The same 400 MB buffer pools if the access patterns are *identical*
- Up to 400 MB buffer pools if the access patterns are *different*

Similarly, in the CF you now have:

- The same 400 MB buffer pool if the access patterns are identical
- The *logical sum* of the local buffer pools if the access patterns are different

Hot Spots

If you dedicate part of the database or transactions to some OS/390 system, then smaller amounts of buffer pools are feasible in the local systems, but not in the CF. The more *hot spots*, the smaller the group buffer pool.

2.2.5.3 DB2 Buffer Pool Sizing Summary

Table 9 on page 74 summarizes the buffer pool sizing guidelines discussed in the previous sections.

Table 9. Parallel Sysplex DB2 Buffer Pool Sizing Summary

Access Pattern	GBPCACHE(ALL)	GBPCACHE(Changed)
Same	$BP_{PS} = BP_{SS}$ $GBP = BP_{SS}$	$BP_{PS} = BP_{SS}$ $0.10 \times \sum_{n=1}^{n} BP_{PS} \leq GBP \leq 0.40 \times \sum_{n=1}^{n} BP_{PS}$ **(1)**
Different	$BP_{PS} \leq BP_{SS}$ $0.50 \times \sum_{n=1}^{n} BP_{PS} \leq GBP \leq 1.00 \times \sum_{n=1}^{n} BP_{PS}$ **(2)**	$\frac{BP_{SS}}{n} \leq BP_{PS} \leq BP_{SS}$ $0.10 \times \sum_{n=1}^{n} BP_{PS} \leq GBP \leq 0.40 \times \sum_{n=1}^{n} BP_{PS}$ **(1)**

Note: Description of terms

Term	Description
BP_{PS}	*Buffer Pool$_{Parallel\ Sysplex}$*
BP_{SS}	*Buffer Pool$_{Single\ System}$*
GBP	*Group Buffer Pool*
n	*Number of DB2s in the Data Sharing Group*

Note: (1)

Factor	Condition
10%	Light sharing with a low amount of updating and/or many “hot spots”
40%	High amount of sharing with a lot of update activity and/or few “hot spots”

Include hiperpool in the summation of buffer pools.

Note: (2)

Factor	Condition
50%	Few table spaces, indexes, or partitions specify GBPCACHE ALL and/or many “hot spots”
100%	Almost all the table spaces, indexes, or partitions specify GBPCACHE ALL for a heavy sharing environment and/or few “hot spots”

Do not include hiperpool in the summation of buffer pools.

2.3 IMS/DB Parallel Sysplex Data Sharing Considerations

In IMS V5 and V6, IMS can share data at the block level across multiple OS/390 images, with up to 32 IMS subsystems. This is called sysplex data sharing. Each OS/390 image involved in sysplex data sharing must have at least one IRLM (IRLM V2.1 and above), and each IRLM must communicate using the OS/390 Cross Systems Coupling Facility (XCF).

This section describes some of the IMS/DB capacity planning considerations when migrating from a single OS/390 image system to a Parallel Sysplex:

- Expected I/O growth
- Parallel Sysplex CPU cost
- Locking rate calculation
- Buffer pool sizing

Four key IMS/DB Data Sharing Publications

For a discussion on performance and capacity planning topics related to IMS/DB Parallel Sysplex data sharing, refer to:

IMS/ESA V6 Administration Guide: System, SC26-8730
OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook, SG24-2076
IMS/ESA V5 Administration Guide: System, SC26-8013
S/390 MVS Parallel Sysplex Performance, SG24-4356

IMS Quick-Sizer Case Studies

For an example on how Quick-Sizer supports IMS/IMS Parallel Sysplex capacity planning, refer to the introduction to Quick-Sizer provided in 6.2, “CP2000 Quick-Sizer Overview” on page 164.

An IMS/DB2 case study is provided in 7.1, “Case Study: DB2 Migration to a Data Sharing Environment” on page 197.

2.3.1 IMS/DB Parallel Sysplex Data Sharing

When multiple OS/390 systems share data across *multiple* OS/390 images (up to 32) and use a CF, it is known as sysplex data sharing.

Each OS/390 image involved in sysplex data sharing must have at least one IRLM (IRLM V2.1 and above), and each IRLM communicates using XCF services. Extended Recovery Facility (XRF) can be used in a sysplex data sharing environment, if the alternate system is in the same sysplex.

IMS sysplex data sharing can be used whether you have IMS TM or CICS as the transaction manager.

IMS/ESA V5 Block Level Sharing

IMS/ESA V5 introduced the capability to share data at the block level across multiple systems.

The data was kept in local buffer pools on each OS/390 system and data coherency was maintained through two cache structures, one for OSAM and one for VSAM. These structures were directory-only and therefore did not contain actual database data, only a list of the subsystems with an interest in the validity of that data.

IMS/ESA V6 Block Level Sharing

In IMS/ESA V6, the OSAM structure was changed so that it now holds data as well as the directory; OSAM structures use the store-through cache algorithm.

This structure might therefore be substantially larger than the corresponding IMS/ESA V5 structure. The VSAM structure is a directory-only structure, like in IMS/ESA V5, and contains no data.

IRLM Lock Structure

IRLM uses a lock structure to provide locking capability among IMS systems sharing data. The entire structure must reside in CF control storage.

Fast Path Databases

IMS/ESA V6 introduced support for n-way data sharing of Data Entry Data Bases (DEDBs) with Virtual Storage Option (VSO) areas and Sequential Dependents (SDEPs). SDEPs only use a Lock structure while VSOs cache data similarly to DB2. DEDB/VSO can have multiple structures for the same area, and you can allocate two cache structures that hold data and put one on each of two CFs.

The support for VSO is implemented by the use of one or two store-in cache structures for each area. The data is initially written to the cache and periodically cast out to DASD. This provides the equivalent function to IMS/ESA V5 where the data was held in a dataspace and cast out from there.

Shared Message Queues and Expedited Message Handler

IMS/ESA V6 can exploit the CF for workload balancing and insured-message delivery. For IMS full function this is basically an all-or-nothing proposition, that is, if you pick to use shared message queues (SMQs) all transactions are run using it. Each IMS/FP message that run remotely will “write, read, delete” from the shared expedited message handling queue (SEMHQ) structure of the CF.

Both SMQs and EMH queues use list structures. The *primary* list structure contains the shared queues. The *overflow* list structure, if defined, contains shared queues that overflow after the primary reaches a predefined (use specified) threshold. Quick-Sizer will assist you in estimating the CPU cost at both the sender CEC and the CF in addition to providing guidance on CF structure sizing associated with the usage of IMS SMQs.

IMS/DB Data Sharing Group

An IMS *data sharing group* includes:

- Databases
- A single set of RECON data sets
- One or more CFs
- A single IRLM lock structure in the CF
- OSAM, VSAM and VSO DEDB cache structures

IMS connects to the structures in the CF. When a database block or VSAM control interval (CI) is updated by one IMS, IMS makes sure all copies of the block or CI in other IMS buffer pools are marked invalid. Buffer invalidation works in all IMS sysplex database environments: DB/DC, DBCTL and DB batch. In these environments, IMS supports buffer pools for VSAM, VSAM hiperspace, OSAM, and OSAM sequential-buffering buffers.

2.3.2 Expected I/O Growth for IMS/DB

The number of I/Os *per transaction* in a single image is a function of local buffer pool hits: the higher the hit ratio, the lower the I/O rate. When moving to a Parallel Sysplex, I/Os per transaction become a function of the local buffer pool hit ratio *and* the global buffer pool hit ratio. The expectation is that local buffer pool hit ratios are slightly lower in a Parallel Sysplex than in a single image system. This is primarily due to buffer cross-invalidations caused by updates on other systems.

Rule of Thumb for Expected Additional I/Os

With IMS as the database manager, the number of misses to local buffer pools are expected to increase by 1% to 5% per system added to the Parallel Sysplex. It averages about 3% per system. This means an average increase in I/Os per IMS transaction of 1% to 5% per system added to the Parallel Sysplex.

CP2000 uses a default value of 3% increase. This is further discussed in Chapter 6, “Using CP2000 for Capacity Planning in a Parallel Sysplex” on page 161.

You may be able to reduce the I/O rate by increasing the buffer pool sizes, but there is a limit to how far this can be taken.

2.3.3 Parallel Sysplex CPU Cost for IMS/DB Data Sharing

CP2000 provides you the possibility to examine the needed capacity of a Parallel Sysplex exploiting IMS/DB2 data sharing. The following formula is used as a rule of thumb.

Note: This formula is identical to the one used for DB2 data sharing as discussed in 2.2, “DB2 Parallel Sysplex Data Sharing Considerations” on page 63.

IMS/DB Parallel Sysplex Data Sharing Rule of Thumb

IMS/DB Parallel Sysplex Data-Sharing Cost Rule Of Thumb:

$$A = M\% \times T + (E\% + ((N - 2) \times I\%)) \times D$$

where:

A = Additional RPP needed to support multisystems management and data sharing in a Parallel Sysplex
T = Total RPP of the OS/390 system you are sizing
D = RPP of the application that is sharing data (IMS/DB)

D includes the following elements:

- RPP for IMS/DB, including its use of OS/390 and VTAM services
- RPP for CICS or IMS/DS, including its use of OS/390 and VTAM service

N = Number of OS/390 images participating in the Parallel Sysplex
M = Multisystems Management cost associated with a Parallel Sysplex (3%)
E = Enablement cost for data sharing in a Parallel Sysplex
I = Incremental growth cost for data sharing in a Parallel Sysplex

For a discussion of the term Relative Processing Power (RPP), refer to 1.3.1, “From Cycle Time to M-value” on page 8. CP2000 does not directly use this formula; however, its calculations are based on the concept expressed by this formula.

Table 8 on page 66 shows the coefficient E (from the formula above) for various Parallel Sysplex configurations. The coefficient for enablement cost of 100% IMS/DB Data Sharing is about the same as for DB2.

The following examples illustrates how this formula works. Let us calculate the additional Parallel Sysplex RPP (A) using this formula on page 77 and the coefficient E from Table 8 on page 66, and the coefficient I for incremental growth mentioned above.

Example 1:
An IMS/DB application is using 100 RPP today. The IMS/DB application uses 50% of the OS/390 system in which it runs. The OS/390 system is running in a non-sysplex environment. Part of the application (20 RPP) is migrated to a Parallel Sysplex and will share IMS/DB data. The CFs used are 9674 C02s. The workload is spread across three OS/390 images each running on 9672-R2 based CECs. Only these three systems share IMS/DB data.

M = 3%
T = 100 RPP/0.50 = 200 RPP
E = 15%
N = 3
I = 0.5%
D = 20 RPP

Applying these numbers and coefficients to the data sharing formula for IMS/DB yields:

$$A = 3\% \times 200\ \text{RPP} + (15\% + ((3 - 2) \times 0.5\%)) \times 20\ \text{RPP} \approx 9\ \text{RPP}$$

Example 2:
An OS/390 image operates as a base sysplex and uses 100 RPP running on a 711-based CEC. Ten percent of the OS/390 system is an IMS/DB application that is migrated to a Parallel Sysplex composed of two 9672-R3 based CECs coupled to a 9674-C03.

M = 0%
T = 100 RPP
E = 15%
N = 2
I = 0.5%
D = 10 RPP

Applying these numbers and coefficients to the data sharing formula for IMS/DB yields:

$$A = 0 \times 100\ \text{RPP} + (15\% + ((2 - 2) \times 0.5\%)) \times 10\ \text{RPP} \approx 2\ \text{RPP}$$

For more details, refer to *S/390 Parallel Sysplex System Performance Considerations Presentation Guide,* which can be requested by IBM representatives as MFTS package on MKTTOOLS.

2.3.4 IMS/DB Locking Rate Calculations

Some of the major factors affecting the performance of a Parallel Sysplex in a data sharing environment include:

- The IMS locking rates
- The lock contention rates
- The database access rates
- The database read/write ratio
- The rate of physical I/O activity

IBM has conducted IMS/DB data sharing benchmarks. CP2000 predicts the locking rates for IMS/DB based on IBM IMS/DB data sharing benchmarks and customer experiences. If you do not think that these benchmarks are representative for your workload, then you need IMS trace data. IMS trace data enables you to better predict how Parallel Sysplex would perform in your environment.

There are two methods of collecting data for obtaining locking rates that may be used to predict performance impact to a Parallel Sysplex:

1. IMS lock trace
2. Lock estimation

Note: If you used IRLM for locking prior to migrating to Parallel Sysplex, then as an approximation the locking rates will be very similar.

If you were using PI-locking, then all this gives is an estimation.

2.3.4.1 Lock Traces

Depending on your IMS configuration, such as:

- CICS/DBCTL
- IMS(TM)/IMS(DB) Full Function (FF)
- IMS Fast Path (FP)

There are different procedures to obtain locking rates. With or without IRLM there are several procedures to run. We will give an overview of the procedures. For more information, your IBM representative may require the document *IMS Guide for Locking Calculations*, available from MKTTOOLS (PTSLOCK PACKAGE).

Running one of these procedures will provide you with locking numbers.

Traces with IRLM

This should be used if IRLM locking is be used. If running data sharing between two systems, the procedure must run on both systems at the same time.

Traces with IMS Program Isolation (PI)

For a single IMS, running a non-data sharing program, isolation locking or IRLM locking may be used. Running this procedure will provide you with an estimation of the locking rate. A PI trace may be heavy and is sometimes difficult to use since it impacts normal operations.

IMS Fast Path

Fast Path has different locking protocols than Full Function. Fast Path does Control Interval (CI) locking only, no record locking. If the database is not shared then, Fast Path uses Fast Path lock manager. If the database is shared, then Fast Path uses IRLM as a lock manager.

Batch Considerations

Often, batch jobs drive the highest locking rates and create the maximum number of locks held in a system. For a significant number of installations, batch processing will create the maximum level of lock activity. The previously described procedures also apply for batch.

If you have not gone through any of the calculations in the list above, you can let CP2000 provide you with a default lock access rate. CP2000 uses the following formula:

$$IMS_{\text{lock access rate}} = 0.25 \times \text{M-value}$$

This is similar to stating that the

$$IMS_{\text{lock access rate}} = 0.80 \times IMS_{\text{I/O rate}}$$

The factor 0.25 is an assumption and relates to experience. So far, variances from 0.15 to 0.30 have been observed. This factor may change over time depending on availability of data. If you think this factor does not reflect your system, feel free to change the lock/unlock access rate number. In Chapter 6, “Using CP2000 for Capacity Planning in a Parallel Sysplex” on page 161, you will

find a discussion on how CP2000 Quick-Sizer as an option allows you to change its default access rate number. Take a quick glance at the Quick-Sizer panel shown in Figure 59 on page 178.

2.3.5 IMS/DB Buffer Pools Sizing in Parallel Sysplex

When data sharing is active, IMS/ESA V6 has local buffer pools in the OS/390 systems as well as globally in the CF. Figure 15 illustrates local buffer pools (BP) and OSAM and VSO DEDB cache structures in the CF. Additional list and lock structures to maintain buffer coherency are located in the CF too.

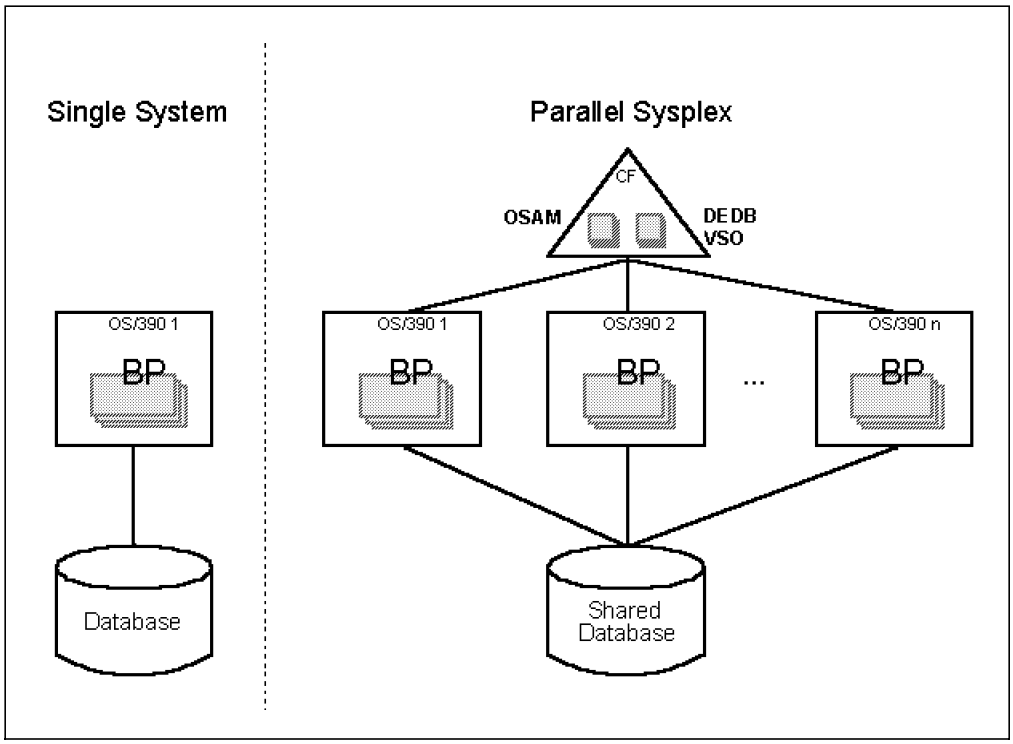


Figure 15. IMS/DB Buffer Pools, Single OS/390 versus Parallel Sysplex

IMS/ESA V6 OSAM Structures

IMS/ESA V6 OSAM structures sizes are determined by you. Their size is related to the sum of the local OSAM buffers and the average of the smallest and largest buffer entry size. The calculation for IMS/ESA V6 OSAM structures differ from IMS/ESA V5. The calculation for IMS/ESA V6 VSAM structures is similar to IMS/ESA V5.

IMS/ESA V5 OSAM and VSAM Structures

IMS/ESA V5 OSAM and VSAM structure sizes are determined by you. Their size is related to the sum of the local (OSAM and/or VSAM) buffers, and the level of repetition of data between systems. Recommended starting values range from 2.0MB (10,000 buffers) to 203MB (1,000,000 buffers), with 37MB (100,000 local buffers, 10% common among four systems) considered a *practical* starting value.

IRLM Lock Structures

The IRLM lock structure size is a function of the number of systems in the IMS data sharing group and the number of lock table entries required (to the power of 2). Recommended starting values range from 1.2MB (1-6 IRLMs, 2^{19} lock table entries) to 4,352MB (22-32 IRLMs), with 32MB (1-6 IRLMs, 2^{23} lock table entries) considered a *practical* starting value.

VSO DEDB Structures

VSO DEDB structure sizes are determined by the VSO DEDB Area definitions specified in the DBDGEN utility.

Shared Message Queues

Shared message queue structures are dependent on the message size, message rate, and the time the message is retained in the queue.

The tool IMSSMSCP available on MKTTOOLS, assist in determining the CF resources needed for shared message queue structures.

OSAM and VSAM structures are discussed for both IMS/ESA V6 and IMS/ESA V5 in the following sections.

2.3.5.1 OSAM and VSAM Structures

With IMS/ESA V6, OSAM structures can be used for caching data in addition to buffer invalidation (as in IMS/ESA V5). VSAM structures remain unchanged and the sizing of these remains as for IMS/ESA V5.

IMS/ESA V6

For OSAM buffers in IMS/ESA V6, the structure size has to be large enough to hold enough directory entries for all buffers and data entries for cached buffers. The actual size of the structure is determined by the SIZE parameter specified in the CFRM policy and the directory element to data element ratio can be specified in DFSVSMXX.

To determine the structure size for IMS/ESA V6 OSAM structures, you may use the buffer pool definitions contained in the IOBF statement of the DFSVSMxx member as input to the formula described in the following.

Recommendation

The recommendation is to always carry out the calculation described in the following and do the specifications to the system as described. If it is not used, the default calculation for the directory-to-element ratio is done and it is rarely appropriate.

IMS/ESA V6 OSAM Structure Sizing Example

Consider two IMS subsystems in your data sharing group IMS1 and IMS2. Each has three OSAM buffer pools consisting of:

- IMS1: 3000 2KB buffers, 4000 4KB buffers, 3000 6KB buffers, for a total of 10000 buffers
- IMS2: 1000 2KB buffers, 6000 4KB buffers, 3000 6KB buffers, for a total of 10000 buffers

Since there is one directory entry for each buffer, there is a total of 20000 directory entries.

For IMS/ESA V5, the structure size is approximately:

Directory Entry Size $\times$ *No. of Buffers*

which in this case would be approximately:

200 *bytes* $\times$ 20000 $\approx$ 4*MB*

In IMS/ESA V6 data entry elements are required in addition to the directory elements. By default IMS calculates the directory-to-element ratio by using the

average number of CF structure elements (defined as 2KB for OSAM) required for the smallest defined subpool size and the number of the elements required for the largest defined subpool size.

The directory-to-element ratio for the example is thus derived as follows:

$2KB\ SP = 1\ Data\ Element\ (Per\ Directory\ Entry)$

$6KB\ SP = 3\ Data\ Elements\ (Per\ Directory\ Entry)$

$Total\ No.\ of\ Data\ Elements = 1 + 3 = 4$

$The\ Average\ No.\ of\ Data\ Elements = 2$

$The\ Directory\text{-}Entry{:}Data\text{-}Element\ Ratio = 1{:}2$

The expression for the total structure size now becomes:

$\{Average\ of\ Smallest\ and\ Largest\ Buffer\ +\ Directory\ Entry\ Size\} \times No.\ of\ Buffers$

So in this example:

$\{(2KB + 6KB)/2 + 200\ Bytes\} \times 20000 \approx 85MB$

In this instance the default calculation has not worked out too badly because on average there are about two 2K elements for the buffers. If we had more 2K buffers and less 6K buffers IMS would perform the same calculation but we would now be short of directory entries.

We find that 42000 data entries are needed in the cache and 20000 directory entries, so we recommend defining SIZE=85 MB in the CFRM policy and CFOASM=(strname,20,42) in the IMS definitions.

2.4 IMS DC Shared Message Queue (SMQ)

Shared message queues provide the mechanism for up to 32 IMS DB/DC and/or DBCTL subsystems to have access to a common set of message queues. There are two types of queues, one for full function and one for Fast Path, both of which are stored as list structures in a CF. There are potentially two structures per queue, a primary and an overflow (which is optional).

IMS does not access the structures directly; this is achieved by the use of services provided by an address space, introduced with IMS/ESA V6, known as the common queue server (CQS).

Implementation of shared message queues is simplified by the basic design of IMS. In general, IMS does not have facilities (such as temporary storage and transient data) that leave data in an address space for later retrieval because it has always used a multiple region architecture.

Full Function Message Queue:: The long and short message queues, which were on DASD in previous releases, are now combined into a single structure, referred to as the MSGQ. Message queues are supported either in a CF structure or on DASD, but you are *not* allowed to have a combination of the two.

To obtain a message from the MSGQ, an IMS registers interest with its local CQS and is notified when a message is placed on the queue. This means that IMS does not have to request work whenever a resource such as a message region is available.

The flow of work through the shared queues group (SQG) is simple in concept, with each discrete element of work being potentially available to any IMS that has registered an interest. Therefore, terminal input, transaction processing, and terminal output could be processed on one system or on three systems depending on their workloads at the time. The package called IMSSMQCP on MKTTOOLS contains a tool to assist you in planning CF resources for the IMS shared message queue structures.

Fast Path Queue:: In previous releases, Fast Path did not use a message queue. IMS V6.1 introduced an optional CF structure, known as the Expedited Message Handler Queue (EMHQ). This is fundamentally the same as the MSGQ, but processing can be directed by exit DBFHAGU0, such that the EMHQ is not used. The exit can select one of three options: *local only, local first,* or *global only.*

Local only This option ensures that work executes on the receiving system. The EMHQ is not used

Local first This option will try to have the work executed on the receiving system. If this is not possible, then the EMHQ is used.

Global only This option always put the message on the EMHQ.

Overflows:: The primary structure is divided into queues which represent resources such as transactions and LTERMs. If the primary structure reaches the percentage in use threshold, defined at CQS startup, then the most active queues may, if permitted by a user exit, be moved to the overflow structure.

Common Queue Server (CQS):: CQS acts as a server to manage shared queues on behalf of a client, in this case an IMS subsystem. It is also available to non-IMS clients. Up to 32 CQSs may connect to one structure, with each CQS having only one client. Each CQS has its own address space and communicates with the other members of the sharing group by using XCF services. Access to the shared queues is via XES services.

CQS is responsible for recovery in the event of a structure or CF failure. This is achieved through a combination of structure recovery data sets (SRDSs), two per structure pair, and a merged log of activity from all sharing CQSs.

The SRDSs are used to periodically checkpoint the queues from their associated structure pairs. Activity between checkpoints is contained in a CQS log stream. If a structure needs to be rebuilt the SRDS is read, and forward recovery is performed using data from the log stream.

2.5 CICS/VSAM RLS Data Sharing Considerations

CICS Transaction Server (TS) is able to distribute its workload across multiple OS/390 and CICS regions to achieve goals such as high availability and good response time. In the CICS/VSAM environment, it is possible to share VSAM databases across multiple AORs on multiple OS/390 images.

CICS - CICS Multiregion R/W Data Sharing

VSAM Record Level Sharing (RLS) is a function provided by DFSMS/MVS V1.3 and exploited by CICS TS. RLS is designed to enable VSAM files to be shared, with full update capability, between many applications running in many CICS regions.

CICS - Batch R/O Data Sharing

With RLS, CICS regions sharing VSAM data sets can reside in one or more OS/390 images within a Parallel Sysplex. RLS provides limited benefits when data sets are being shared between CICS regions and batch jobs.

VSAM RLS removes the need for CICS file-owning regions (FORs). Instead, DFSMS/MVS supports a new data sharing subsystem, SMSVSAM, to provide the RLS support required by CICS AORs and batch jobs. The SMSVSAM server uses the Coupling Facility (CF) for its cache and lock structures. It also supports a common buffer pool for each OS/390 image. All buffers and control blocks for RLS files reside in the SMSVSAM data space.

Files opened in RLS mode can be accessed by existing CICS application programs without modification.

CICS/VSAM RLS Integrity Options

CICS supports three options to provide full read integrity under the RLS function:

No Read Integrity

With this option, there is no read integrity (so-called Dirty Read) and SMSVSAM does not perform any locking functions. It may be possible to see uncommitted data.

Consistent Read

With this option, a request to read a record is queued if the record is being updated by another task. The read completes only when the update is complete and the Unit of Work (UOW) has released its exclusive lock. In this environment, there is no access to uncommitted data.

Consistent Read Explicit

With this option, the processing for the read request is the same as for consistent read requests. However, the UOW holds its shared lock until syncpoint. This ensures that a record read in a UOW will not have been modified if the task repeats the reads. This is useful when a series of related read requests are issued that have to be consistent and unmodified until the last record is read.

The first two options can be used for files accessed by CICS and batch. The third option is valid only for CICS.

The various integrity options affect the sharing of the data between CICS and batch. Before talking about the different scenarios, we introduce a new concept for VSAM files: VSAM files accessed in RLS mode can be defined as “recoverable” or “non-recoverable.”

Recoverable, Non-Recoverable Attribute

A recoverable data set is one that participates in syncpoint activity and is eligible for changes to be backed out in the event of a transaction failure. When several changes are made to recoverable data sets from a unit-of-work, these changes are either all committed or all backed out.

A non-recoverable data set is one for which updates are committed immediately, and so cannot be backed out. Therefore, there is no backout recovery, and neither CICS nor VSAM provide any support for forward recovery logging.

Depending on whether a VSAM file is recoverable or not, we can have different scenarios while sharing data between CICS and Batch:

- Recoverable data sets
 - Recoverable data sets being updated concurrently by CICS regions can be read by batch jobs with full data integrity.
 - Batch jobs cannot update recoverable data sets when the data sets have been opened in RLS mode by CICS regions, because batch jobs do not perform logging. For CICS regions, logging is done by CICS using the services of the OS/390 system logger. The only way a batch job can update the data set is by first quiescing all CICS activity to that data set, and then running the batch job, using non-RLS access to the data set.
- Non-recoverable data sets

 There are no update restrictions for non-recoverable data sets. Both CICS and batch can read and update them.

VSAM Transactional Services

A proposed enhancement to DFSMS, going beyond RLS support, is called VSAM Transactional Services (TS). With these extensions, it becomes possible for CICS and Batch jobs to have full read/write sharing capabilities for recoverable data sets. This would include logging and forward recovery. Batch applications may require modification to inform VSAM TS when a syncpoint has been reached and a set of related updates should be committed.

Announcement of this function will be based on IBM′s business and technical judgment. This information represents IBM′s current intent and is subject to change or withdrawal.

Capacity Planning

In order to be able to achieve availability by spreading your workload across multiple AORs in multiple CECs, you should reorganize your CICS configuration in a multiple region (TOR-AOR) design first. Capacity planning also includes the cost of implementing a TOR-AOR-FOR setup. This cost applies to other data sharing environment including IMS/DB and DB2 using CICS as the transaction monitor.

If you have a TOR-AOR-FOR setup today, and replace the FOR with an SMSVSAM address space and RLS mode files, you can size the database environment and take the cost of the new data sharing environment into consideration.

If your starting point today is a single CICS region and all access to VSAM files is local (no function shipping of requests to an FOR), you will see more overhead due to data sharing and dynamic transaction routing.

The following section describes some of the VSAM RLS capacity planning considerations that are important when migrating from a single OS/390 image system to a Parallel Sysplex.
We discuss the following topics:

- MRO sizing
- Expected I/O growth
- Parallel Sysplex CPU cost
- Locking rate calculation
- Buffer pool sizing

VSAM RLS Data Sharing Publications

For further discussions on performance and capacity planning topics related to VSAM RLS Parallel Sysplex data sharing, refer to:

- *CICS and VSAM Record Level Sharing: Implementation Guide*, SG24-4766
- *S/390 MVS Parallel Sysplex Performance*, SG24-4356-02

2.5.1 MRO Considerations

The RLS data sharing environment is designed to support sharing of VSAM files among multiple cloned AORs. The existence of multiple AORs provides the potential for continuous availability and operation to the installation. Before approaching and sizing your RLS data sharing environment, you should size the CICS MRO environment. Figure 16 summarizes the potential configuration and provides an idea of the cost of MRO communication and routing between the different CICS regions. For information on how to use CP2000 to size the cost of MRO, refer to 6.2.1.1, “Quick-Sizer Input Definitions” on page 166. IBM benchmarks suggest an increase of about 16% to 20% in the CPU time used by the single region application. This increase is due to function shipping between the AOR and the FOR. Response times usually do not change significantly.

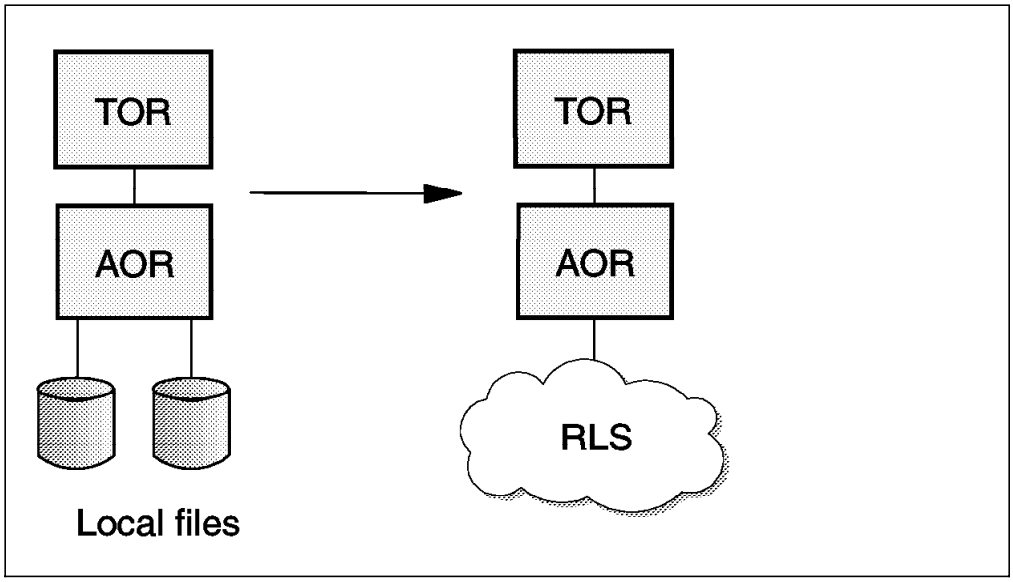


Figure 16. CICS Configuration Using Local Files versus CICS/VSAM RLS

2.5.2 VSAM RLS Parallel Sysplex Data Sharing

CICS has for many years provided sharing of VSAM data via “function shipping,” in which all access to a data set was done through a single CICS region, the File-Owning Region (FOR). This also improves availability in the event of user code causing CICS abends in that other AORs and FORs are not affected by a single AOR abend.

SMSVSAM allows the CICS AORs to access the data set directly, eliminating the need for an FOR and avoiding problems of bottlenecks or single points of failure.

SMSVSAM further improves data availability and can increase the total throughput of the Parallel Sysplex.

VSAM RLS Data Sharing

RLS data sharing is performed by SMSVSAM acting as a server. SMSVSAM resides in a separate address space and processes the RLS requests. There is one server per OS/390 image in the sysplex.

SMSVSAM requires at least three Sharing Control Data Sets (SHCDSs), to keep all the data sharing related information. A set of initialization parameters is contained in the IGDSMSxx parmlib member.

Before starting the SMSVSAM server, you should also allocate the following structures in the Coupling Facility:

RLS CF Structure Usage

Lock structure There is one lock structure for all the SMSVSAM servers. It is used to maintain integrity across multiple servers.

Cache structures These structures are used as shared buffer pools for the VSAM data. They are used to cache data that is of interest to more than one SMSVSAM server and to maintain data consistency across local buffer pools by using a cross-invalidate (XI) mechanism. We recommend allocating at least two cache structures on different Coupling Facilities in order to get fast recovery in case of a Coupling Facility connectivity failure.

2.5.3 Expected I/O Growth for RLS

No Expected I/O Growth for CICS/VSAM RLS

The number of I/Os per transaction is dependent on the local buffer hit ratio. The buffer pool hit ratio is dependent on the type of workload (read/write ratio). To avoid growth in I/Os when adding systems, you should scale the buffer pool and Coupling Facility cache size to maintain a constant hit ratio in the buffer pool. If the cache size is set up corrrectly, one should experience a drop in the DASD I/O rate per transaction. If the Coupling Facility cache is sized large enough to avoid reclaims, the XIs due to updates will not cause an increase in the I/O rate because of an increase in the Coupling Facility cache request rate. Because the RLS buffermanager manages its buffers more efficiently than local shared resources (LSR) pools, the buffer pool hit ratios will be better for the RLS managed buffers than for LSR pools.

Benchmarks with a 1-way CICS/VSAM RLS workload split to run in a 2-way and 3-way data sharing configuration show no growth in the I/O rate. The number of XIs is dependent on the workload, but you can assume that an I/O growth of zero is fairly accurate if a CF cache of a reasonable size is being used. If a small CF

cache is being used (in relation to the buffer pool size), the amount of data that is cached in the CF decreases and more I/O occurs.

You can readjust the CF cache size to obtain a good I/O hit ratio in the buffer pool based on information in the SMF 42 subtype 19 record.

2.5.4 Parallel Sysplex CPU Cost for RLS Data Sharing

CP2000 Quick-Sizer supports capacity planning for a Parallel Sysplex exploiting VSAM RLS data sharing.

The following formula is used as a rule of thumb.

Note: This formula is identical to the one used for DB2 and IMS/DB data sharing as discussed in 2.2, "DB2 Parallel Sysplex Data Sharing Considerations" on page 63.

VSAM RLS Parallel Sysplex Data Sharing Rule of Thumb

VSAM RLS Parallel Sysplex data sharing cost rule of thumb:

$$A = M\% \times T + (E\% + ((N - 2) \times I\%)) \times D$$

where:

A = Additional RPP needed to support multisystems management and data sharing in a Parallel Sysplex.

For a discussion of the term Relative Processing Power (RPP), refer to 1.3.1, "From Cycle Time to M-value" on page 8.
T = Total RPP of the OS/390 system you are sizing
D = RPP of the application that is sharing data (CICS regions)

D includes the following elements:

- RPP for CICS TORs, including its use of OS/390 and VTAM services
- RPP for CICS AORs and FORs, including its use of OS/390 and VTAM services

N = Number of OS/390 images participating in the Parallel Sysplex
M = Multisystems management cost associated with a Parallel Sysplex (3%)
E = Enablement cost for data sharing in a Parallel Sysplex
I = Incremental growth cost for data sharing in a Parallel Sysplex

Note: RPP and other capacity measures such as MIPS and M-value are discussed in 1.3, "Capacity Planning and CPU" on page 8. CP2000 does not directly use this formula; however, its calculations are based on the concept expressed by this formula.

OS/390 V2R4 provides enhancements to the Type 6 SVC to improve performance to the VSAM path taken by CICS. This fix is also available to prior releases of OS/390 starting with MVS/ESA V5.2. This should increase your ITR about 3%.

Table 8 on page 66 shows the coefficient E (from the preceding formula) for various Parallel Sysplex configurations. The coefficient, for the enablement cost of 100% VSAM RLS data sharing, is about the same as for DB2.

The following examples illustrate how this formula works. Let us calculate the additional Parallel Sysplex RPP (A) using the formula and the coefficient E from Table 8 on page 66, and the coefficient I for incremental growth.

Example 1:

A CICS VSAM application is using 100 RPP today. The CICS application uses 50% of the OS/390 system in which it runs. The OS/390 system is running in a non-sysplex environment. Only some of the application's AORs and FORs (20 RPP) will be migrated to a Parallel Sysplex and will share VSAM data. The CFs used are 9674 C02s. The workload will be spread across three OS/390 images, each running on 9672-R2-based CECs.

M = 3%
T = 100 RPP/0.50 = 200 RPP
E = 15%
N = 3
I = 0.5%
D = 20 RPP

Applying these numbers and coefficients to the data sharing formula for VSAM RLS yields:

$$A = 3\% \times 200\ \text{RPP} + (15\% + ((3 - 2) \times 0.5\%)) \times 20\ \text{RPP} \approx 9\ \text{RPP}$$

In this case most of the cost is the multisystem management cost of sysplex enablement.

Example 2:

An OS/390 image operates as part of a two-system sysplex and uses 100 RPP running on a 711-based CEC. Ten percent of the OS/390 system is used by a TOR-AOR-FOR CICS application that is migrated to a Parallel Sysplex composed of two 9672-R3-based CECs coupled to a 9674-C03.

M = 0%
T = 100 RPP
E = 15%
N = 2
I = 0.5%
D = 10 RPP

Applying these numbers and coefficients to the data sharing formula for VSAM RLS yields:

$$A = 0 \times 100\ \text{RPP} + (15\% + ((2 - 2) \times 0.5\%)) \times 10\ \text{RPP} \approx 2\ \text{RPP}$$

For more details, refer to *S/390 Parallel Sysplex System Performance Considerations Presentation Guide,* which can be requested by IBM representatives as MFTS package on MKTTOOLS.

2.5.5 VSAM RLS Locking Calculations

Some of the major factors affecting the performance of a Parallel Sysplex in a RLS data sharing environment include:

- The RLS locking rates
- The lock contention rates
- The integrity options chosen to access the files
- The database access rates
- The database read/write ratio
- The rate of physical I/O activity

Controlling the locking rate is a fundamental step in achieving optimal performance.

In the case of RLS, this can be accomplished by carefully choosing the various read integrity options, ranging from NRI (no read integrity), to CR (consistent read), and CRE (consistent read explicit). The NRI option is equivalent to the level of integrity provided today, and is recommended for applications that can use it.

It should be noted that there is an increased risk of deadlocks when a read integrity level other than NRI is used.

The general recommendation is to use the NRI option. This avoids locking overhead while returning a current copy of the record. It is true that for a recoverable file, the returned record could be the new version of an updated record and the update could subsequently be backed out. While the CR option would not return the uncommitted version of a record, the fact that the record lock is released before the application processes the returned record means the record may be changed or deleted before it is processed by the application. The CRE option ensures consistency of the record across the life of the transaction that issued the GET CRE (CICS repeatable read) request. Where a higher level of integrity is required, it is recommended from a performance perspective to go with the CRE option. In other words you may want to consider CRE for your files if your application needs CR.

The decision to use NRI, CR, or CRE is an application design consideration. In cases where NRI is sufficient, it offers a performance advantage.

Recommended Default Change

DFSMS is responsible for deadlock detection for files opened in RLS mode. The DEADLOCK_DETECTION parameter in IGDSMS in SYS1.PARMLIB specifies the intervals for local and global deadlock detection. The default for this value is (15,4). You may want to change these values to (1,1) to minimize the time required to detect a deadlock condition. Obviously there is a trade off between time spent detecting deadlocks and the effect that deadlocks are causing to applications. Carefully monitor these effects in your environment before making changes to the supplied defaults.

Lock Structure Monitoring

You may use the D SMS,CFLS command to get information on the RLS Lock structure (IGWLOCK00) usage. The output give the false contention rate, lock rate, and related information.

Lock Structure Sizing ROT

You can use the following simple rule of thumb formula to size the lock structure for VSAM RLS data sharing:

$10MB \times number_of_systems \times lock_entrysize$

where:

number_of_systems

is the number of systems in the sysplex.

lock_entry_size

is the size of each lock entry. This value depends on the MAXSYSTEM value that is specified in the sysplex couple data set.

MAXSYSTEM **Value**	**Lock Entry Size in Bytes**
7 or less	2
>= 8 and < 24	4
> = 24	8

For further details, refer to *DFSMS/MVS V1.3 DFSMSdfp Storage Administration Reference*, SC26-4920. For large installations in need of more accurate sizes, use the tool RLSLKSZ from MKTTOOLS.

You can monitor the performance of your Coupling Facility lock structure with the SMF 42, subtype 17 records and the Coupling Facility RMF Report to check for the false contention value; you should adjust this value to below 0.1% by increasing the size of the lock structure.

2.5.6 VSAM Cache Usage and Sizing in a Parallel Sysplex

VSAM files are assigned to cache structures through SMS definitions. These cache structures function as an intermediate level of storage between the local buffer pool and DASD. For capacity planning you have to consider sizing both Coupling Facility cache structures and the SMSVSAM local buffer pool.

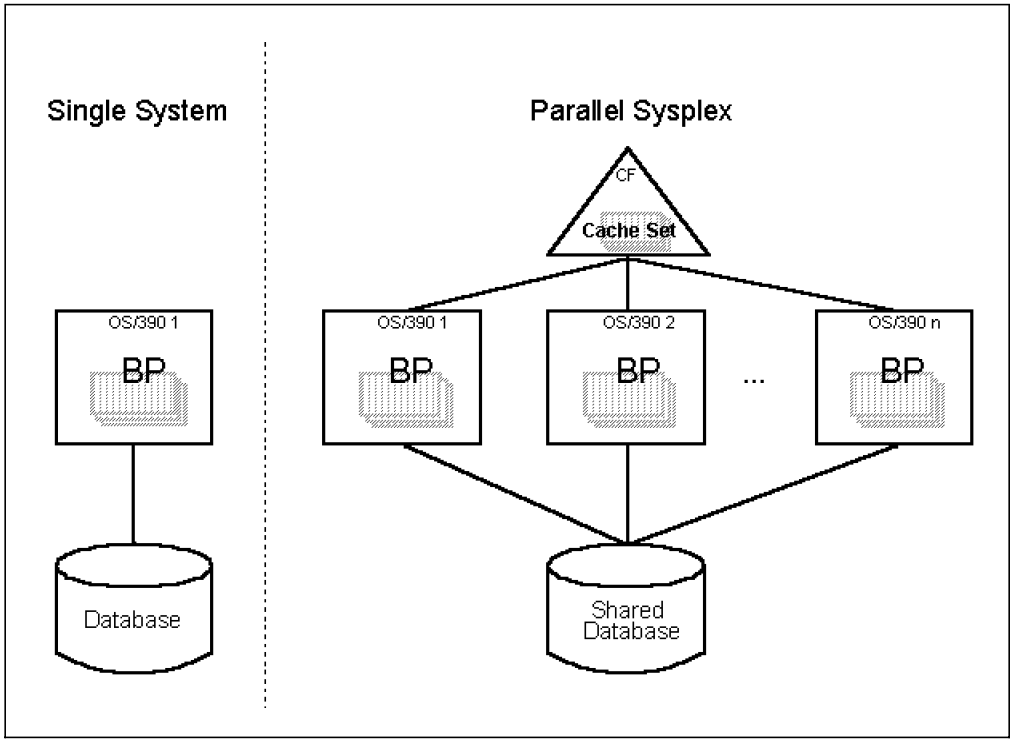


Figure 17. CICS Cache Set, Single OS/390 versus Parallel Sysplex

Cache Set Buffer Pool Structure

As shown in Figure 17, RLS data is held in two places: in the CF cache structure and in a local buffer maintained by the SMSVSAM server.

A request to read data from an RLS file results in a copy of the CI containing the record requested being placed both in the local buffer of the SMSVSAM server on the OS/390 image from which the request was made, and also in the CF cache structure. This CF cache has both directory and data elements. The data elements are used to hold the CIs. The directory is a directory for the cache structure where the system keeps control information about data items shared among cache users. There is one directory entry for each data item that users share. Data items will be stored both in the local buffer pool of the system performing the operation and in the cache structure. If a data item is changed, its directory element provides the connection with the other SMSVSAM servers that have a registered interest in that CI. This enables cross-invalidation (XI) processing to take place.

Storage in both the local buffers and the CF cache has to be managed. This is done in both cases using a “least recently used” algorithm. The two algorithms differ slightly, but basically when there is no storage left to satisfy a request, the storage used by the data that has not been referenced for the longest time is freed. For the CF cache, directory elements are given preference over data elements. This is because the directory elements are used to signal changes to any locally-held CIs to other SMSVSAM servers that have copies of this CI (for XI processing). If there is a demand for cache space, it is better to keep directory elements (rather than data elements) in the cache, because removing directory elements requires that the CIs whose registration information they contained are invalidated.

Any changed CIs will be written to the cache as part of store-through processing, so they should be available to any other servers that need to refresh their copies as a result of XI processing.

If the amount of data held in the local buffers is much larger than that held in the CF cache structure, this causes a lot of CF cache reclaims. There would not be enough space in the CF cache to hold entries for all the CIs held in local buffers, so reclaim would frequently be required.

2.5.6.1 VSAM Cache Structure Sizing

Cache Sizing ROT

The general rule for the CF cache structure is to make its size equal to the size of the hiperspace and LSR (Local Shared Resource) pools in the pre-RLS environment.

SMSVSAM organizes the cache structure with directory elements and data elements. The splitting of the structure storage between directory and data is established by SMSVSAM and altered in order to keep the local hit ratio consistent with different cache sizes by giving up data elements in favor of directory entries when the cache is small. This ratio depends on the demand for CF elements, whether directory elements are favored over data elements, and the size of the CIs that you have in the data elements. Each data element is 2K in size; so the CI size will dictate how many CIs can be held in the data elements.

You can verify your size by examining the Coupling Facility Activity RMF report.

2.5.6.2 VSAM Buffer Pool Sizing

Buffer Pool Sizing ROT

The RLS_MAX_POOL_SIZE parameter in the IGDSMSmm PARMLIB member specifies the maximum size in megabytes of the SMSVSAM local buffer pool.

Laboratory testing has shown that for a workload with a 5.5:1 read/write ratio, a RLS_MAX_POOL_SIZE value twice the size of the LSR pool size was needed in order to maintain the required hit ratio when moving from MRO to RLS. This is the ROT for calculating your local buffer size. Setting this parameter too low may result in unnecessarily degrading the local hit rate. Data obtained from the SMF 42 subtype 19 record can be used to evaluate the local buffer hit rates.

Currently there is no such information in any RMF reports. SMF is the only source for the correct sizing.

2.6 CICS TS Logging Considerations

A brief introduction to the new elements in the CICS TS logging environment follows, offering some confidence with the planning of required resources.

CICS TS Logging Environment

CICS TS has two log streams that constitute the CICS system log, unlike the DFHJ01A and J01B datasets which existed in previous releases.

DFHLOG

DFHLOG is the primary log stream, which is written out to the CF and will contain data for in-flight transactions that may have to be backed out.

DFHSHUNT

The DFHSHUNT log stream is for units of work for which there has been no activity for the last two activity keypoints, or for which some condition has occurred requiring that they be “shunted” aside for later resolution.

The notion of a separate log stream for shunted units of work enables CICS to do “log tail deletion” and keep only the minimum amount of data in the primary log stream. As units of work are completed, and log data is not to be kept, it can be deleted from the primary log stream. Shunted units of work could take an indefinite amount of time before resolution.

Log of Logs

A log of logs is kept, which can be used by products such as CICSVR or other vendor products.

This log can assist in performing tasks such as forward recovery by indicating which logs contain data for which files.

User Journal

User journaling need not go to the MVS Logger unless forward recovery logging has been specified.

User journals can still be directed to SMF. The overhead of writing to SMF is expected to be less than that for using the MVS Logger.

Figure 18 shows the new logging environment and the types of logs.

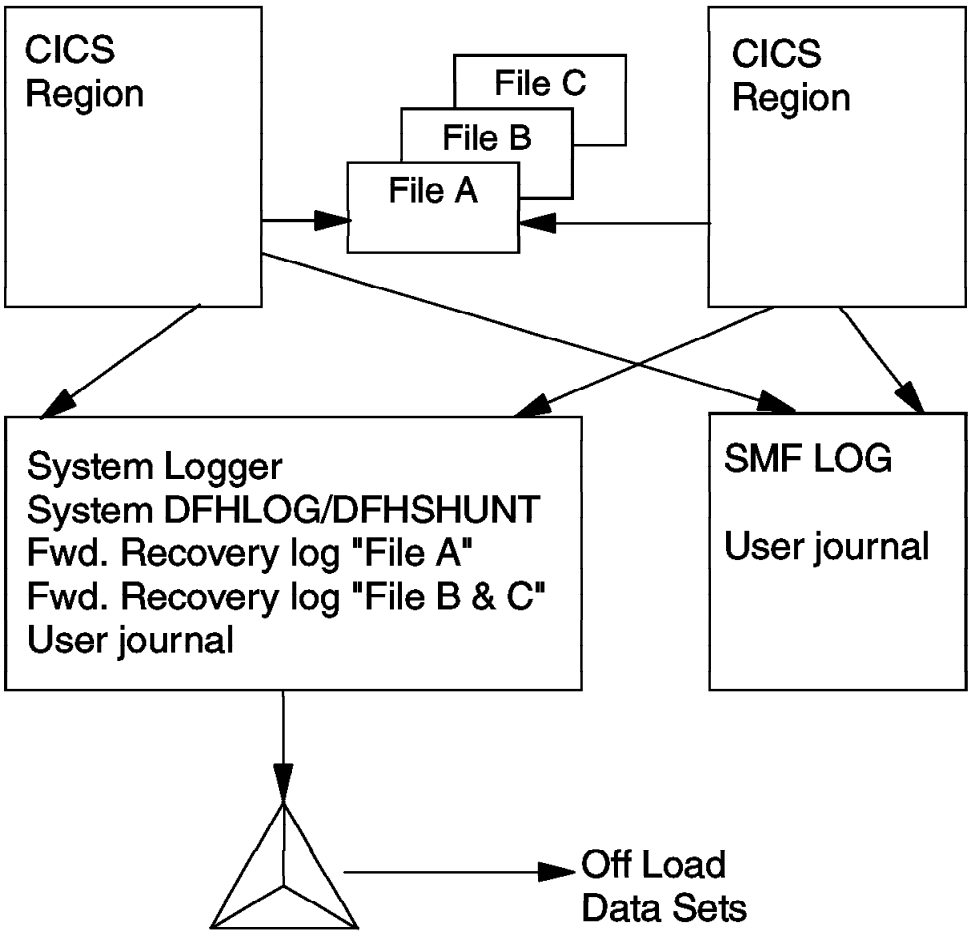


Figure 18. CICS TS Logging Environment

CICS TS V1.1 uses the OS/390 System Logger for all logging and journaling requirements. Using services provided by the logger, the CICS log manager supports two types of logging: the CICS system log, used for dynamic transaction backout, emergency restart and re-synchronizing in-doubt units-of-work; and the forward recovery logs used to recover the files.

Since the OS/390 logger uses the CF to merge data coming from multiple images, you have to plan the CF storage required to support the logging environment. The first capacity planning task is to plan your CICS logging environment in order to be able to accurately size the CF storage. This task includes consideration of the following:

- The number of CICS systems in your environment
- The number of user journals and autojournals you require
- The number of forward recovery logs you require
- The expected average buffer sizes and transaction rates in your environment
- The number and size of CF structures you require
- Your logstream naming conventions, and the number of log stream models that you need to define

Only when you have defined your logging environment, the relation between logstream and CF structures, and the logstream grouping, can you plan your CF requirements.

CICS TS Sizing Utility

CICS TS provides the DFHLSCU utility to assist you in calculating the amount of CF space you need and the average buffer size of your logstreams. (Starting with OS/390 V1R3, this becomes less of an issue). DFHLSCU helps you in determining the following:

- AVGBUFSIZE - the average buffer size of your logstream. It is important that this value accurately reflect the real size of most log blocks that are written to the structure. To a great extent, it controls the efficient use of space within the structure, and can prevent undue DASD off-loading.
- INITSIZE - the initial amount of space to be allocated for the structure in the CF.
- SIZE - the maximum size of the logstream structure.
- STG_SIZE - the size of the staging data set required by the logstream.

If you cannot use DFHLSCU because you do not have CICS/ESA 4.1 or Version 3 journal records to use as input to it, the following sections may help you calculate your space requirements.

In order to avoid wasting CF space, we recommend that you do not place the primary (DFHLOG) and secondary (DFHSHUNT) logstreams in the same structure, because of the large disparity in data volumes written to the primary and secondary system logs.

When planning structure sizes, ensure that you allow enough storage to avoid having the logstream associated with the CICS system log spilling to DASD. Generally, the volume of data that CICS keeps in the primary system log at any one time covers from two to three activity keypoints. This volume is determined by the activity keypoint frequency, which is measured in blocks of CICS log data, and defined on the AKPFREQ system initialization parameter. Review the number of blocks specified on the AKPFREQ system initialization parameter when planning CF structure sizes.

For further details, refer to *CICS TS for OS/390 CICS Performance Guide*, SC33-1699.

2.6.1 DFHLOG Sizing

The system log CF structures are much larger than those used for other logstreams and have the greatest potential impact on system performance, so it is important to try to size the structures accurately so that you do not waste CF space by overallocating, nor cause poor performance by underallocating.

The basic Coupling Facility space requirement for each DFHLOG is:

```
CFspace = x * AKPFREQ * avgbufsize / 1024   K bytes per logstream
(round avgbufsize up to a multiple of 256)
where x=
1.25 if SYSLOG=KEEP is specified
3.3 if SYSLOG=NOKEEP is specified
```

You can calculate AVGBUFSIZE for DFHLOG from the weighted average of the data logged by the most frequently executed transactions in the system:

```
AVGBUFSIZE = bytepersec / (writepersec + 48)
```

where:

- bytepersec = (N1 * D1) + (N2 * D2) + ... + (Nn * Dn)
- writespersec = lesser of 33 or ((N1 * D1) + ... + (Nn * Dn)), where:
 - N1, N2 Nn are the transaction rates per second of the most frequently executed transactions.
 - D1, D2 Dn are the bytes of data logged by each transaction.

You can calculate the amount of data (Dn) written to the system log for each transaction, as follows:

```
Dn = Ns * syncreclen +
     Nfc * (fcrechdr + fcreclen) +
     Nts * (tsrechdr + tsreclen) +
     Ntd * (tdrechdr + tdreclen) +
     Nur * (urrechdr + urreclen)
```

where:

- Ns is the number of syncpoints per transaction (usually 1)
- syncreclen is the syncpoint record length
- Nfc, fcrechdr, fcreclen are the number of recoverable updates made, the length of the record headers (usually 144), and the length of the record for file control, respectively.
- Nts, tsrechdr, tsreclen are for recoverable temp storage. For PUT records, tsrechdr is 108, and tsreclen is 88. For UPDATE records, tsrechdr is 108, and tsreclen is 52.
- Ntd, tdrechdr, tdreclen are for recoverable transient data updates. Tdrechdr is 108 and tdreclen is 380.
- Nur, urrechdr, urreclen are for user records written to DFHLOG. Urrechdr is 125.

Since system log CF space requirements are so large, we do not need to reserve 20% of the space for handling spikes in the load; so rather than using

the default highoffload of 80%, we could set HIGHOFFLOAD(95) and aim to delete an AKPs data when the structure reaches 90%.

2.6.2 DFHSHUNT Sizing

It is very difficult to predict the frequency and blocksize for records written to DFHSHUNT. Normally these should be sufficiently small and infrequently used so that accurate planning is not essential.

A rough estimate of values to choose is as follows:

- Avgbufsize = 4096
- Cfspace = 150K per logstream

2.6.3 Forward Recovery Logs Sizing

You can merge the forward recovery logs written by many CICS regions onto the same logstream. You can also use the same logstream for forward recovery data for multiple data sets.

Use the following formula to size the CF space:

```
CFspace = no.entries * avgbufsize / 1024   K bytes per logstream
(round avgbufsize up to a multiple of 256)
```

where no. entries = writespersec * 6.5.

Determine writespersec = lesser of 33 or (N1 + ... + Nn)

where N1 ... Nn is the transaction rate per second writing to each data set.

You can calculate AVGBUFSIZE as follows:

AVGBUFSIZE = bytespersec / (writespersec + 36), where:

- bytespersec = (N1 * Wr1 * (D1 + rechdr) +.. (Nn *Wrn *Dn+rechdr)))
- writespersec = lesser of 33 or (N1 + ... + Nn), where:
 - N1 ... Nn is the number of transactions per second writing to each data set.
 - Wr1 ... Wrn is the number of write requests per transaction.
 - D1 ... Dn is the average record length for each data set.
- rechdr is the record header length of each record.

Forward recovery logs should only contain file control records. These can be DATA SET NAME records, which consist of a 204-byte record header, and no further data.

Alternatively, the records may be WRITE ADD, WRITE ADD COMPLETE, or WRITE ADD DELETE records. In this case, rechdr is 84 and is followed by the record key and the record data (including its key).

2.6.4 User Journal and Autojournal Sizing

You can use some formulas used to estimate the CF space for estimating the forward recovery logs.

When log blocks are not forced to a log, the average block size will tend to be slightly less than the MAXBUFSIZE value defined for the CF structure.

For journals where log records are forced, the average block size can be calculated from the weighted average of the data logged for each of the journals being logged to the same logstream for one CICS region, and the number of WAIT requests.

2.6.5 Monitoring the Logger Environment

CICS collects statistics on the data written to each journal and logstream, which can be used to analyze the activity of a single region. However, since logstreams can be shared across multiple OS/390 images, it is often more useful to examine the statistics generated by OS/390.

The OS/390 system logger writes SMF 88 records containing statistics for each connected logstream. It also supplies a sample reporting program, IXGRPT1, in SYS1.SAMPLIB, which can be used as is, or modified to meet your requirements. IXGRPT1 does not report the average buffer size, but this can easily be found from "bytes written′/′# writes completed" to verify the estimates made during setup. This information can also be found from CF structure statistics that are available from the RMF CF activity reports.

The main item that should be monitored routinely is the number of "Structure full" events. If these occur frequently, it means that the logger cannot write data to DASD fast enough to keep up with incoming data and that CICS has to wait before it can write more data. To resolve such problems:

- Increase the CF structure size to smooth out spikes in logger load.
- Reduce the data written to the stream by not merging so many journals or forward recovery logs onto the same stream.
- Reduce the Highoffload threshold percentage.
- Examine device IO statistics for possible contention on the DASD packs used for logstream data sets.
- Use faster DASD devices.

For system logs, with SYSLOG=NOKEEP, the best performance is achieved if CICS deletes the logs′ data when no longer needed before they are written to DASD by the OS/390 logger.

2.7 CICS TS Temporary Storage

Temporary storage data sharing allows your CICS applications to access non-recoverable TS queues from multiple CICS regions running on any OS/390 image within a Parallel Sysplex. CICS uses temporary storage data sharing to store shared queues in a structure in a CF, access to which is provided by a CICS temporary storage server address space.

Figure 19 on page 99 shows the CICS TS temporary storage environment.

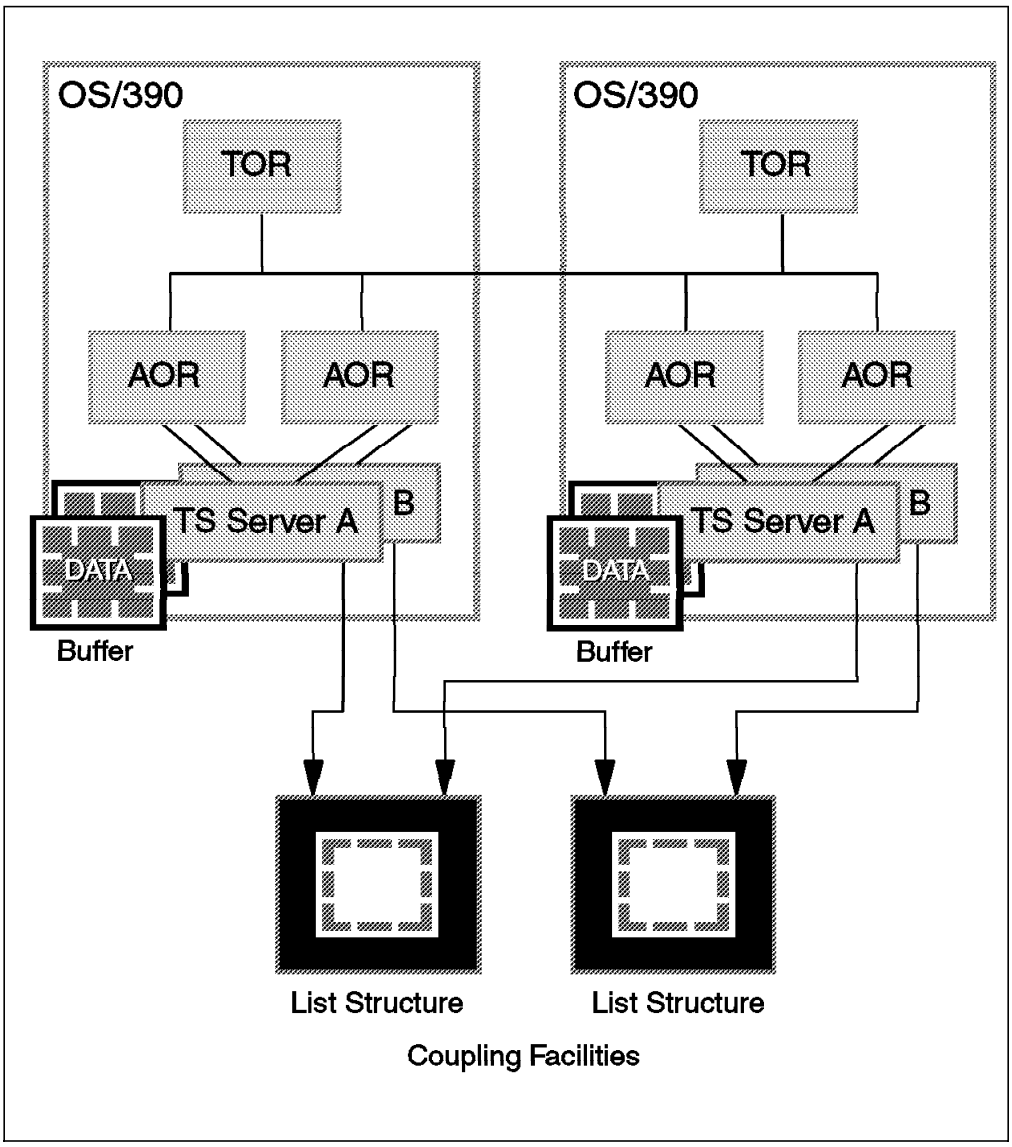


Figure 19. CICS TS Temporary Storage

CICS stores a set of TS queues that you want to share in a TS pool. Each TS pool corresponds to a CF list structure defined in the CFRM policy.

CICS TS Description

You can create a single TS pool or multiple TS pools within the sysplex, to suit your requirements. For example:

- You could create separate pools for specific purposes, such as a TS pool for production, or a TS pool for test and development.
- You could create more than one production pool, particularly if you have more than one Coupling Facility and you want to allocate TS pool list structures to each Coupling Facility.

From a capacity planning point of view, you should evaluate two resources:

- MRO/XCF TS Queue Owing Region versus TS data sharing ITR
- Coupling Facility storage required to support TS data sharing

2.7.1 CICS TS ITR Considerations

During your capacity planning activity, you should consider which is the best CICS TS sharing configuration in order to get a good ITR. The choice of using TS data sharing versus a TS Queue Owing Region accessed from AORs via function shipping depends on the type of workload and the TS usage and queue length. An IBM Benchmark was run to compare the two environments.

TS is faster than QOR if mixed workload

For a typical TS user (approximately 25% of the code path length of each transaction is TS operation and the rest is application code), TS data sharing offers better throughput over a QOR. TS data sharing CPU usage is lower (7%), and transaction response times are equivalent.

ITR considerations

For a workload whose transaction code is 100% TS operation, we have different results, depending on the length of the TS queue.

The following workloads are to be considered as exceptions:

- Installations with short queues (individual queue items less than 4K in length and overall queue lengths of less than 32K) have better throughput using CICS TS data sharing. This is because all of the TS operations versus the Coupling Facility are treated as synchronous. CPU usage is lower (9%) and transaction response times are equivalent.
- Installations with long queues (individual queues more than 4K in length and overall queue lengths more than 32K) have better performance with a QOR configuration. CPU for CICS TS data sharing will increase about 20% versus QOR. This is because all the Coupling Facility accesses are asynchronous and hence more expensive in terms of response time.

2.7.2 TS Coupling Facility Sizing

TS Coupling Facility Sizing

If you plan to use CICS temporary storage sharing, you also have to plan for storage in the Coupling Facility.

To size the Coupling Facility storage for TS pool, use the following formula:

```
Item entry size = (170 + (average item size, rounded up to next 256))
                  + 5% extra for control information

Total size = 200K +
             (number of large queues x 1K) +
             (number of items in all queues x item entry size)
```

For further details on CICS temporary storage, refer to *CICS TS for OS/390 System Definition Guide*, SC33-1682

Chapter 3. OS/390 UNIX Services Sizing and Capacity Planning

At a Glance

OS/390 UNIX Services provide a totally new application programming interface to the traditional S/390 environment. Specifically, it is now possible to integrate applications that were originally established for completely separated UNIX platforms, with the robust, safe, and highly available operating system OS/390.

To do this, you will immediately face the question, *"what will it cost me to integrate this UNIX application on OS/390?"* In this chapter we try to answer this question. Furthermore we describe what OS/390 resources are specifically used by OS/390 UNIX Services.

Recommended Reading:

- *Accessing OpenEdition from the Internet*, SG24-4721
- *OpenEdition MVS Installation and Customization*, SG24-4529
- *OS/390 R3 OpenEdition, Installation and Customization*, SG24-2087 (not yet available)
- *OpenEdition MVS for MVS/ESA 5.1 Presentation Guide*, GG24-4095
- *Porting Applications to the OpenEdition MVS Platform*, GG24-4473
- *Selecting a Server - The Value of S/390*, SG24-4812
- *The Cost of Scalability*, ITG International Technology Group `http://www.s390.ibm.com/marketing/costs.html`

3.1 Introduction

This chapter is written from the viewpoint that the best place to run UNIX applications is on S/390. We do not discuss issues of porting; we assume that the decision has been made to run on S/390. There are many studies which show that the total cost of computing is lower on OS/390 systems than on UNIX systems, and so we expect many customers to implement new UNIX applications such as SAP R/3 and Lotus Domino on S/390.

© Copyright IBM Corp. 1998

OS/390 is UNIX95-certified

OS/390 UNIX Services is the implementation of UNIX as defined by X/Open, within OS/390. The first release of OS/390 UNIX Services was called MVS OpenEdition and was included in MVS/ESA V4.3; it added functions defined by POSIX 1003.1, 1003.1.a, 1003.2, 1003.2a. Subsequently, OS/390 UNIX Services was enhanced until it became a full-function UNIX as defined by X/Open. This was achieved in OS/390 V1.2, which obtained UNIX95 branding in 1996.

With the inclusion of UNIX Services, new application programming interfaces and commands are also integrated into OS/390. These functions and services provide a UNIX interface to OS/390 in addition to the non-UNIX application programming interfaces. More importantly, existing applications continue to run as before.

Before we discuss sizing for UNIX Services, we explain some differences in the way capacity planning is typically done for UNIX and S/390 systems.

3.2 The UNIX Methodology

The UNIX world has several scales by which the power of UNIX machines is compared. The figures most commonly given are those provided by System Performance Evaluation Cooperative (SPEC), and those defined and certified by Transaction Processing Performance Council (TPC). There are other benchmarks, such as Dhrystone and SPECmark-F. For completeness, however, in Figure 20 on page 103 we show the most well-known benchmarks with respect to their type of measurement.

The figure is organized so that the x-axis indicates to what extent a benchmark is CPU-oriented, whereas the y-axis represents the I/O orientation. Whether a workload containing a relatively high I/O content is more complex than a workload containing a low I/O content is debatable. From the S/390 perspective, a workload containing a high I/O content, including database access and data sharing, will be viewed as more complex while others could say that a workload containing large portions of CPU processing with relatively little I/O is more complex.

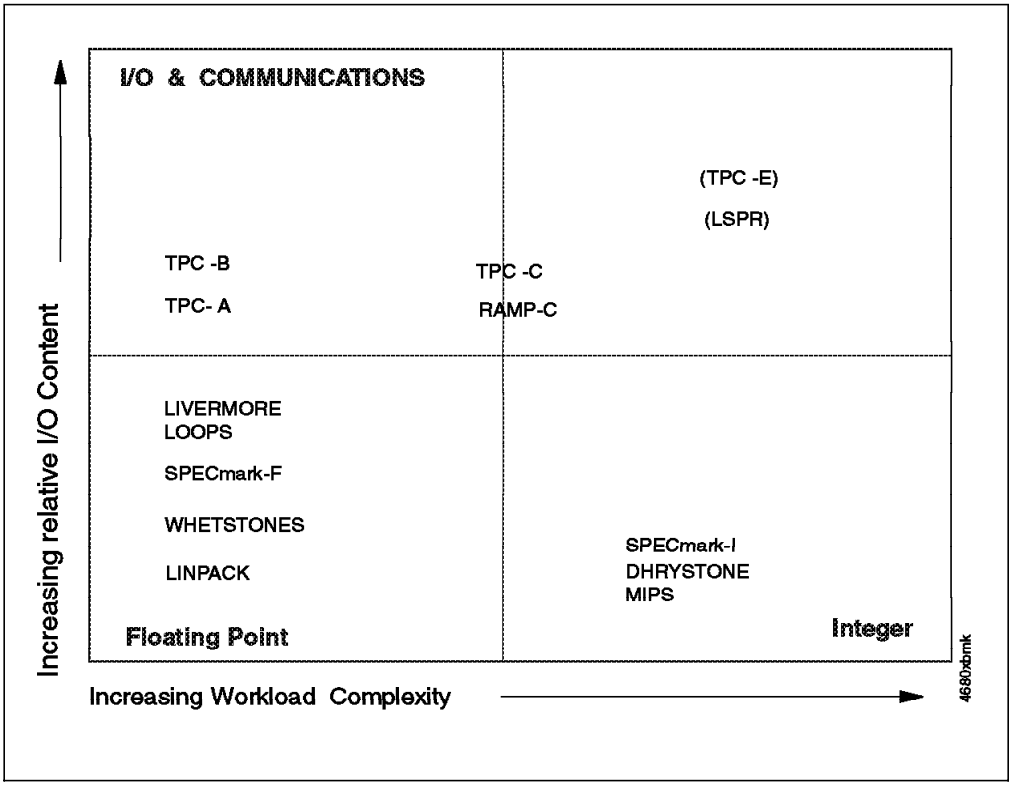


Figure 20. Typical UNIX/PC Benchmarks

The most prevalent numbers used in a commercial environment are those based on TPC's definitions. As TPC itself is a group where UNIX system and PC system vendors are a majority, their definitions are dominated by what the vendors find to be representative.

As you can see, there is only one definition in the upper right corner of the picture, TPC-E. TPC-E was defined as a mixed workload very similar to the way we plan capacity on S/390 (LSPR). An example of a mixed workload is one that contains both online and batch work. Unfortunately, the TPC-E definition was never released by TPC. For completeness, we have indicated where to position Large Systems Performance Reference (LSPR). Notice that both LSPR and TPC-E represent similar types of workload.

Further Information

If you need detailed benchmark data, look at the BENCH package. Your IBM representative can access it from one of the following sources:

- BENCH PACKAGE on MKTTOOLS
- http://ramjet.austin.ibm.com/bench.html
- anonymous ftp from ramjet.austin.ibm.com under /pub/bench4_4

3.2.1 The SPECint Benchmarks

SPECmark-F is also referred to as SPECfp and SPECmark-I is also referred to as SPECrnt.

The following quote is from the SPEC home page on the Internet (`http://www.specbench.org/spec/`):

"The System Performance Evaluation Cooperative (SPEC) was founded in 1988 by a small number of workstation vendors who realized that the marketplace was in desperate need of realistic, standardized performance tests. Their key realization was that an ounce of honest data was worth more than a pound of marketing hype."

SPEC Benchmarks are CPU- and Memory-(Cache) Intensive

The SPEC benchmarks attempt to measure the power of the engine as an indicator of performance by measuring CPU-intensive kernels. While CPU speed has a first order effect on capacity, it is not the only factor and, as we shall see later, there is a considerable range in the balance between I/O and CPU for the various machines. The SPECint benchmark measures infinite cache performance (in the sense that there is minimal effect from processor L2 cache misses) with minimal operating system services. Thus the benchmark workload is very different from typical commercial workloads. From the S/390 perspective SPECint is useful when evaluating progress in CPU design, compiler design, and UNIX Services pathlength, but not really relevant for capacity planning purposes. Also, SPECint is useful in the S/390 perspective for establishing a lower bound of our power relative to UNIX machines.

There were some abuses of the SPECint 92 guidelines that SPECint 95 seeks to prevent. In SPEC terms, UNIX systems often make use of "single user mode" capability to improve benchmark performance. In this case, the service that is responsible for dispatching several processes is turned off. Since SPEC was founded by workstation vendors, this is not surprising. OS/390 does not support single user mode because it is inappropriate for its marketplace. While there is some I/O in the benchmark, the use of disk caching removes physical I/O handling from the benchmark. While the use of processor storage to reduce physical I/O is important, all S/390 installations have very significant I/O rates. In summary, SPEC is a very unrealistic measurement for commercial workloads.

3.2.2 The TPC-C Benchmark

TPC Tries to Model the Real World

The following quote is from the TPC home page on the Internet (`http://www.tpc.org`):

"The TPC is a non-profit corporation founded to define transaction processing and database benchmarks and to disseminate objective, verifiable TPC performance data to the industry."

Several such benchmarks have been defined by the TPC organization. The most current definitions are the TPC-C and TPC-D benchmarks.

Approved in July of 1992, TPC benchmark C is like TPC-A in that it is an online transaction processing (OLTP) benchmark. However, TPC-C is more complex than TPC-A because of its multiple transaction types, more complex database, and overall execution structure. TPC-C involves a mix of five concurrent transactions of different type and complexity, either executed online or queued

for deferred execution. The database comprises nine types of records with a wide range of record and population sizes. TPC-C is measured in transactions per minute (tpm).

TPC-C simulates a complete computing environment where a population of terminal operators execute transactions against a database. The benchmark is centered around the principal activities (transactions) of an order-entry environment. These transactions include entering and delivering orders, recording payments, checking the status of orders, and monitoring the level of stock at the warehouses. While the benchmark portrays the activity of a wholesale supplier, TPC-C is not limited to the activity of any particular business segment, but rather represents any industry that must manage, sell, or distribute a product or service.

3.2.3 The TPC-A Benchmark

Because it was the first definition made by the TPC, we give a short abstract on the TPC-A.

The TPC-A benchmark specifies a response time criterion rather than a CPU load criterion. Thus it would be possible to change the TPC-A rating for a processor by changing the I/O configuration. A further drawback of the TPC-A benchmark was to include the cost of the terminal; that is, the cheaper the terminal was, the better the TPC-A rating became. As these terminals were very simple, teletype-like video display units, it was never possible to compete with these values when using 3720s.

3.3 The Difference between TPC-C and Reality

The TPC-C benchmark allows UNIX systems to show unrealistic results. To a certain extent, the benchmark has also influenced the design of the most recent UNIX systems. Since S/390 was developed to metrics that are independent of this benchmark, its behavior is markedly different and it lacks some of the tuning capabilities that allow the best TPC-C results.

TPC-C is useful for scaling OLTP when the OLTP workload has *all* the following characteristics:

- CPU-bound
- Has partitionable data with affinity to work and users
- Has a very constant rate of work from users
- Has relatively few users (<750)
- Has a relatively small amount of data per user
- All users run the same size and shape work (single application)

An OLTP workload *not* adhering to these conditions will lead to better scaling for S/390 than TPC-C would indicate. Consultants such as the Gartner Group have already commented on this publically and have also made capacity estimates in this context. Refer to 3.6.1, “The Bridge Model” on page 116 for information about IBM-completed measurements establishing equal capacity between a S/390 CMOS 9672-R12 processor and a RS/6000 59H processor. The Gartner Group estimates that TPC-C would show one CMOS 9672-R12 processor equal to one half of a RS/6000 59H processor in capacity at 100% utilization.

We now compare the characteristics of single-workload and mixed-workload benchmarks on UNIX systems to workloads in S/390 environments by considering graphs of response time versus load (Figure 21 on page 106).

- For mixed workloads, UNIX systems exhibit a ramp-like response time as load is increased. For single applications, UNIX systems can be tuned to run very well to higher utilizations.
- OS/390 tends to run the same in either mixed or single application environments, exhibiting more overhead at low utilization and a sharper knee at high utilization breakdown.
- The crossover points are workload dependent.

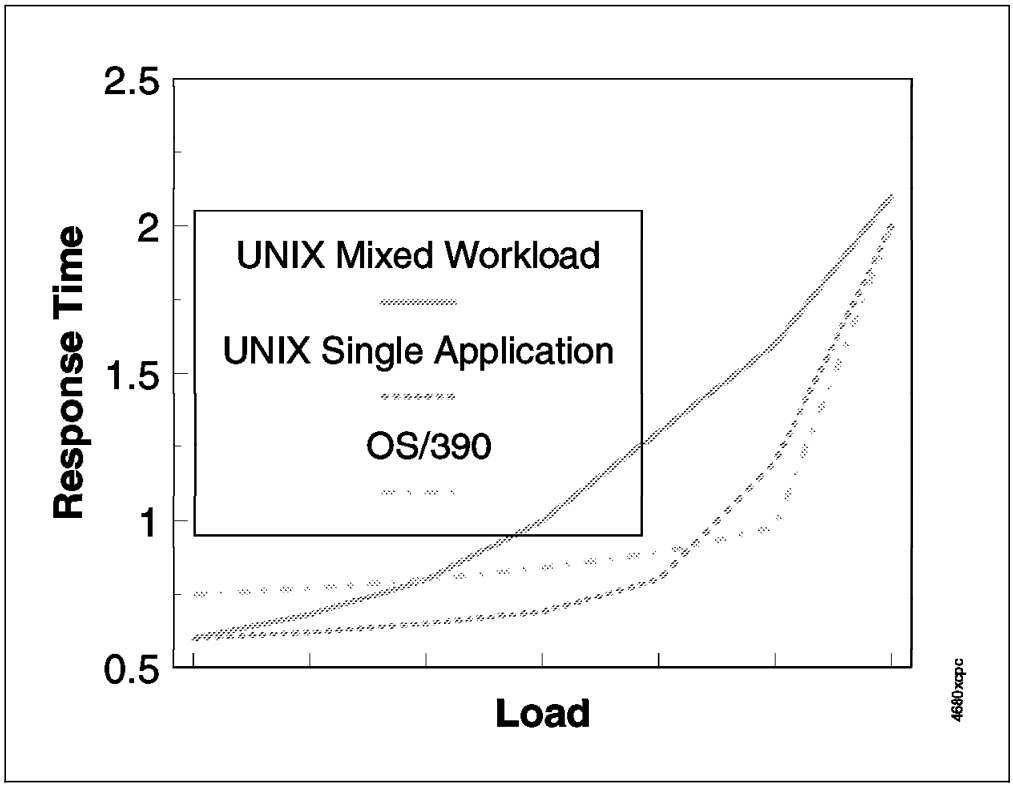


Figure 21. The Capacity Curve

TPC-C is nevertheless a useful metric in the UNIX world because it does allow basic comparisons between systems. Note that TPC-C specifies the processor, operating system and database and vendors are at liberty to choose the database that will produce figures in the best light for their processor.

3.3.1 Comparison of Benchmarks

In Table 10 on page 107 we will show a comparison of the various benchmarks.

Table 10. Comparison of Different Benchmarks

	Comparison Criterion				
Benchmark					
	Any processor?	**Constrained**	**Response metric**	**Tuned**	**CPU utilization**
TPC-A	Yes	Yes	Yes	Yes	Immaterial
TPC-C	Yes	Yes	Yes	Yes	Not directly relevant
TPC-D	Yes	Yes	Yes	Yes	Not directly
SPEC	Yes	No	Yes	No?	At 100%
LSPR	S/390 and compatibles only	No	No	No	Vital

Note: Specification of TPC benchmarks available from `http://www.tpc.org`.

3.4 Application Integration in a UNIX Environment

Figure 22 shows the different approaches to running multiple applications on UNIX systems and on OS/390 systems.

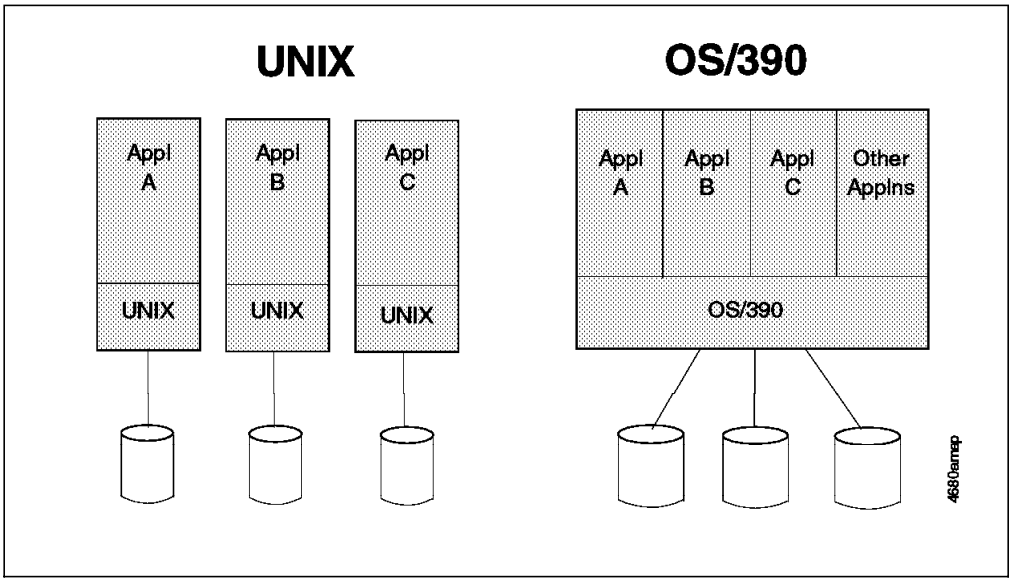


Figure 22. Application Integration in the UNIX Environment

Since there are no facilities in UNIX to allow two applications on a single server with different goals to share resources according to installation-defined specifications, the pragmatic recommendation is that each application run on its own dedicated server. This may be a reasonable choice when there is no business connection between the applications. However, there is a severe impact on operational costs when there is a need for data interchange, data extraction for reporting, and large batch systems. Separate UNIX systems can result in:

- An inability to balance workload across systems, leading to sub-optimal use of processing power

- Interference between batch and online, which often means that batch cannot be run concurrently with online
- Batch job streams that cannot complete within their allotted overnight processing windows
- Complex processes to aggregate data from applications on different servers and the associated systems management problems they cause

S/390 servers, on the other hand, typically support many different types of applications, both online and batch, concurrently. S/390 servers are therefore well placed to service the commercial processing needs of user enterprises.

The higher level of integrated function in OS/390 often means that there is a higher visible entry level cost for software. UNIX software may seem less expensive.[6] However, there are frequently substantial costs later when the enterprise attempts to reproduce the OS/390 level of service with multiple layers of software tools. In smaller situations, where the higher level of service may not be required, other platforms, such as the AS/400 or RS/6000 (UNIX), may be more appropriate.

3.4.1 Workload Management across a Cluster

This section extends the workload management discussion to a clustered environment. Figure 23 compares a *shared everything* approach (as used in S/390 Parallel Sysplex, for example) to a typical *shared nothing* partitioned database approach. It shows the different workload balancing achievable in both tuned benchmarks and real workloads.

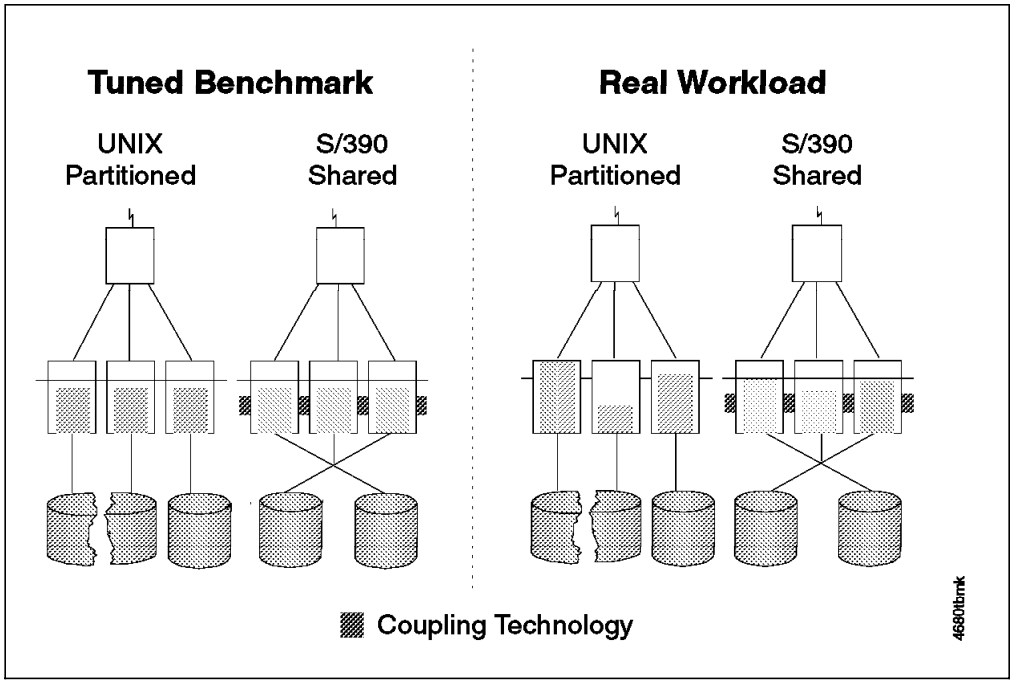


Figure 23. Workload Balancing across a Cluster

Partitioned: In the partitioned, or shared nothing, approach, the database is physically partitioned between a number of servers (three, in this simple

[6] In fact, when you look for TPC results, you will see that in most cases the operating system on the server is free of charge.

example). Each incoming transaction is routed to the appropriate server by a front-end router, which could reside on one of the servers. Each transaction is processed against the database on the selected server only. If there is a need for records residing on one of the other servers, then that request must be routed to the appropriate server. Provided that the volume of such cross-system requests is small, performance will be satisfactory.

Shared: In the shared everything approach, all servers have access to the entire database. Incoming transactions are routed to one of the nodes, based on which node currently has the lightest workload. Since all nodes can access all database records, there is no need for remote data access. Thus workload is dynamically balanced across all systems. On S/390 the coupling technology, operating system, database and transaction manager software combine to provide high performance workload balancing with data integrity.

Tuned Benchmark: For tuned systems, as commonly encountered in benchmarks such as TPC-C, it is relatively simple to partition the database so that the workload on each processor is balanced. In such systems the database partitioning may not be into equal sizes, because it is often found that a large part of the transaction activity is to a very small part of the database. Thus a tuned system could have database partitions of greatly varying sizes because the benchmark has "static" hotspots.

Real Workload: In practice, with most real partitioned systems, the originally tuned and partitioned database will result in very unevenly balanced processor loads. Any attempt to retune by changing the partitioning structure (a disruptive activity) will probably be defeated by the following day′s transactions. This can result in some systems being overloaded, while others are under-utilized. In contrast, the shared everything database of the IBM Parallel Sysplex is not dependent on data reference patterns. Workload activity is automatically and dynamically balanced.

The result of uneven processor utilizations in the partitioned case is less effective use of the hardware resources. This means that you need to buy more resources than for the equivalent shared everything design.

3.5 OS/390 UNIX Services: the OS/390 Perspective

One of the main design points for implementing UNIX Services on OS/390 was not to reinvent the wheel. Wherever possible, UNIX Services are based on functions provided by the base control program of OS/390.

The central hub for OS/390 UNIX Services is the OMVS address space. Most of the application programming interfaces are serviced in this address space: a call to a UNIX service finally ends in a program call (PC) instruction. When an application requires a UNIX service the first time, its address space connects *automatically* to OMVS and gets a new set of UNIX-type control information, mainly process ID (pid), user ID (uid) and group ID (gid). This process is called "dubbing." You will see this address space on the list of OMVS if you issue a `D OMVS,A=ALL` command. The address space is now a UNIX process.[7]

[7] Note: when using the PC instruction, the CPU time is accumulated to the originating address space and not in the OMVS address space.

3.5.1 Creating New Processes

Address spaces in an OS/390 environment are created by TCAS, APPC, SUBMIT, and START commands. In a UNIX environment, there are two system calls that create new processes: spawn(), and fork().

The spawn() process is similar to batch submit. The application starts a new process with only a loose connection to itself, that is, pipes are often used. The program running in the new process is defined by a parameter of the spawn() function.

The Domino-Go Web Server uses spawn() to start new processes on behalf of Common Gateway Interface(CGI)- initiated programs.

The fork() process basically *replicates* the current process into a child process. After the fork(), both processes are nearly identical with respect to the program running and the resources used (open files, socket connections). Both the parent process and the child process continue as independent processes. The child process normally continues for a few lines of code only and then restarts with an exec(). This is a typical UNIX system call. It can be compared with the XCTL macro, except that exec() exchanges the complete program environment at a jobstep task level.

The fork() function requires both CPU and storage on all platforms, including UNIX. “Cloning” means that a copy of nearly all resources, storage, and so on, of the originating address space is made. You can see this in Figure 24.

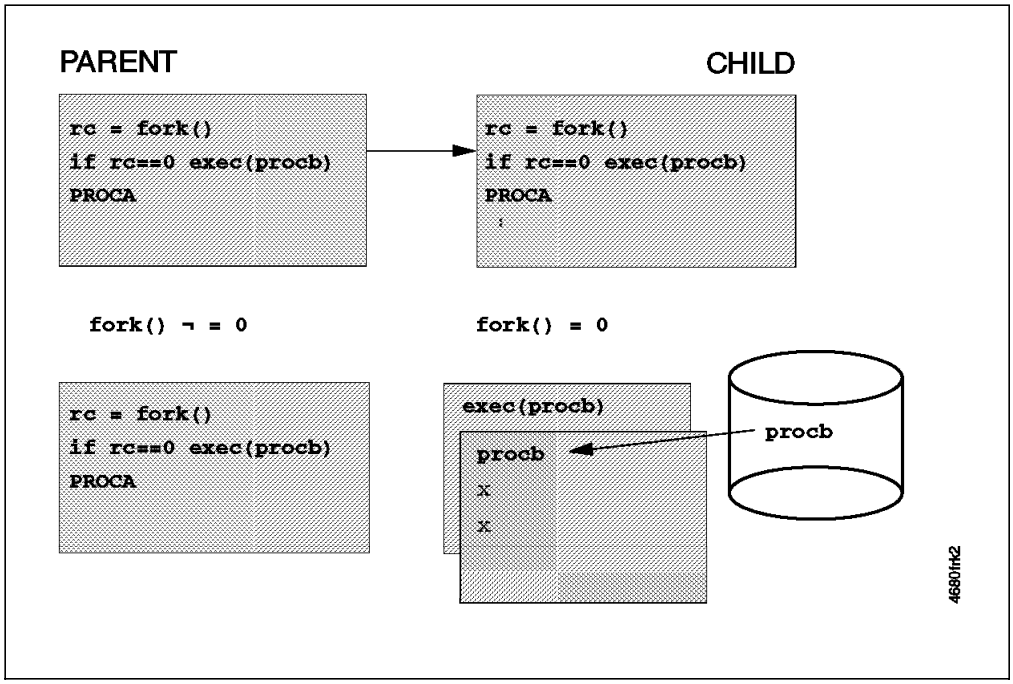


Figure 24. Schematic Overview of Fork()

The child process continues with exactly the same comparison statement as its parent. The inetd daemon, which is included in OS/390 UNIX Services, uses

fork(). The inetd daemon is a central network service in UNIX. It is responsible for monitoring one or more TCP/IP ports[8]

Several services are handled by inetd itself. Other services, such as otelnetd (not to be confused with TN3270), remote login (rlogin), remote shell (rsh), to name a few, are started by inetd using fork() and exec(). This child process is the other end of the client/server relationship.

The processes fork() and spawn() produce new address spaces. You can avoid this by customizing OS/390 UNIX Services in an appropriate way, refer to *OS/390 OpenEdition Planning*, SC28-1890. In case this cannot be done, you have to calculate for more address spaces in an *interactive environment* where the UNIX shell services are used. The rule of thumb is:

```
3 additional address spaces/interactive user
```

An interactive user is a user that calls the UNIX shell from TSO/E, rlogin, or Telnet.

3.5.2 Threads

UNIX threads can be compared to OS/390 subtasks and, consequently, allow for multitasking. OS/390 and its predecessors have provided multitasking capability using task control blocks (TCBs) and attach/detach assembler macros for some time. PL/1 and C have their own APIs to provide multitasking to the application programmer, also based on TCBs. Threads are another API to provide multitasking.

There are three kinds of threads:

1. Light-weight threads

 Light-weight threads are not assigned to individual subtasks, but are managed within one task. This is also sometimes referred to as pseudo-subtasking. OS/390 UNIX Services does not implement light-weight threads.

2. Medium-weight threads

 With medium-weight threading services, the program allocates a pool of OS/390 subtasks to be used for threads. When a new thread is required, one of the free subtasks within the pool is used. If no subtask is available, thread creation waits until a subtask becomes available.

 With medium-weight threading, it is important to monitor how often thread creation waits, because there are no available subtasks. If it happens often, it might be a good idea to increase the size of the subtask pool that is created by this application.

 An example of a UNIX Services application that uses medium-weight threading is the OS/390 Domino-Go Web Server. One of the configuration parameters of this server is the number of subtasks in the pool (the MinActiveThreads directive).

3. Heavy-weight threads

[8] A port is the address that identifies a TCP/IP application. In combination with the IP address, it identifies a specific server instance on a specific host.

If the application uses heavy-weight threads, a new OS/390 subtask is created every time the application requests a new thread to be created. The subtask is terminated when the thread terminates.

Threads are typically used in online environments where a thread is used to handle user- or connection-specific requests. Attach/Detach/SVC 3 processing is done at thread creation and termination.

3.5.3 UNIX File System Usage

To implement UNIX file services, a new file system was designed according to OS/390 standards and UNIX system needs. This was accomplished through the implementation of a hierarchical file system (HFS). This file system can only be accessed using the appropriate UNIX I/O services. It is not possible to access HFS files using traditional access methods (QSAM, BSAM, etc) directly.[9] HFS is based on the Partitioned Dataset Extended (PDSE) function. Therefore, its basic blocksize is 4KB, and when a UNIX Services application reads/writes to HFS, the EXCP rate is derived from this value.

Note: A UNIX Services application can easily use the traditional OS/390 access methods. This option should be considered when heavy sequential file usage is expected.

3.5.4 Services Executed Using PC to OMVS Address Space

The functions necessary to provide the UNIX Services in OS/390 are implemented in a new address space, the OMVS address space, which is automatically started at IPL time. To reach these services, a small stub is linked into the UNIX Services application. This stub finally issues a program call (PC) to the OMVS address space (Figure 25 on page 113).

[9] The NFS client function provides this capability.

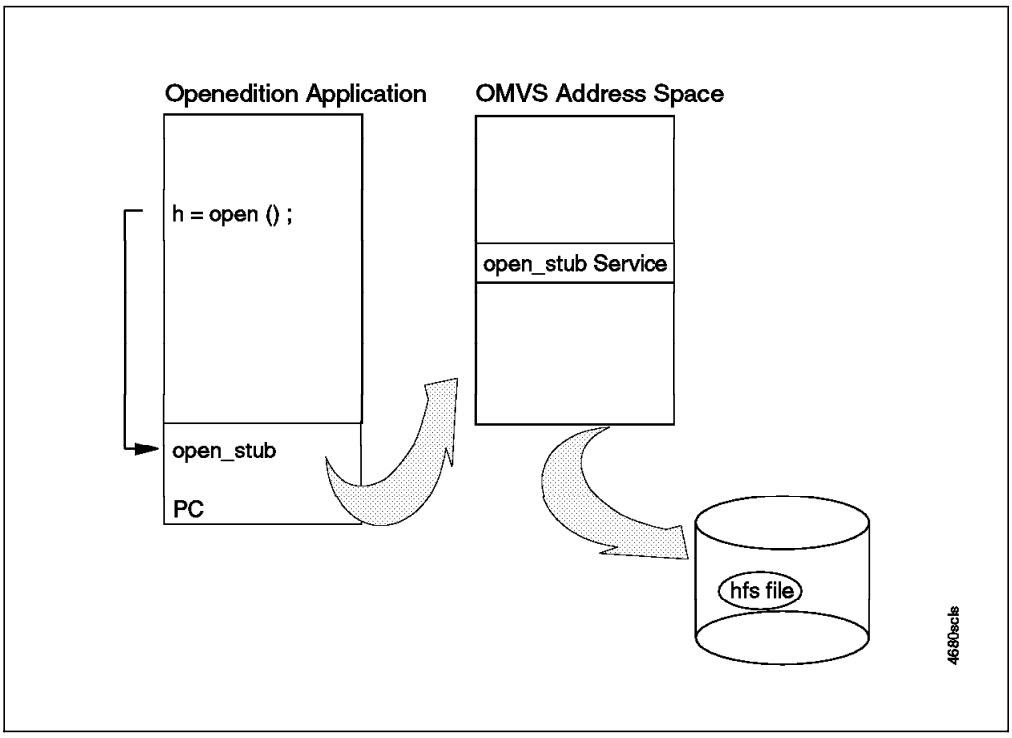


Figure 25. Schematic Overview of Program Call

3.5.5 Pathlength Considerations

An application that uses OS/390 UNIX Services is still an OS/390 application in the sense of using OS/390 services directly or indirectly.

Assuming an application written in C or C++, the application is compiled and linked using the AD/Cycle C Compiler or the C/C++ Compiler for OS/390. Thus the core still behaves as an OS/390 application. The new calls are the UNIX Services, and they are imbedded as callable services, again a typical OS/390 approach. There is *no* dependency on the fact that UNIX operating systems are normally implemented on RISC-type processors.

OS/390 system services are indirectly used or can be directly accessed if needed. For example, when opening a file in the HFS, RACF or another security product is invoked to check the user′s access authority. The UNIX application on OS/390 has benefited from a normal OS/390 service.

When an application is ported from UNIX to OS/390, the current source machine is usually a RISC-based system. Compilers tend to produce smaller code for RISC-based processors than for OS/390, but a lot of service functions are called by the compiled code on the RISC system. Packed decimal data, which are highly used in a commercial environment, are typically handled by RISC machines using callable services, but S/390 has direct support for it. (Actually, as the RISC processor does not necessarily have instructions to handle packed decimal data, it has to *emulate* these instructions with its reduced instruction set.) So the pathlength may be, in the end, of similar size on both types of platforms even though OS/390 has larger object code.

3.5.6 Memory Usage

Due to its implementation, OS/390 UNIX Services allocates storage mainly in the Extended Common Service Area and Extended System Queue Area for several services such as fork(), ptrace(), shmat(), and mmap().

The following gives an overview of when and how much storage is allocated due to the UNIX Services.

3.5.6.1 Extended Common Service Area Usage

Extended Common Service Area storage is directly impacted by the number of processes and the number of tasks (threads) you are going to use in your system:

```
120 bytes per thread used
500 bytes per process used
```

For example, if you plan to support 500 processes and 2000 threads, UNIX Services will consume 490KB.

For other Extended Common Service Area usage, refer to *OS/390 OpenEdition Planning*, SC28-1890.

3.5.6.2 Extended System Queue Area Usage

The largest contribution to Extended Common Service Area usage comes from the usage of shared memory. Shared memory is needed by the functions shmat(), mmap(), fork(), and ptrace(). The amount of shared memory allocated is defined through the MAXSHAREPAGES parameter.

The basic formula for calculating the amount is:

```
number_of_pages * number_of_connections * 32bytes / page
```

Shared memory is typically used by a server and one or more client address spaces. Therefore:

```
number_of_connections = number_of_clients
                        + 1 for the anchor block
                        + 1 for the server address space
                        + 1 for OMVS address space
```

For example, if 500 clients use 8MB of shared memory, the amount of Extended Common Service Area used is:

```
(500 + 1 + 1 + 1) * 32bytes/page * 8MB * 256pages/MB = 33MB ECSA
```

Memory mapping is used by a single process to map a file into storage. Therefore, using the preceeding formula, we will need for 5MB:

```
(1 + 1 + 1) * 32bytes/page * 5MB * 256pages/MB = 123KB ECSA
```

To speed up the fork() function, the parent's page is captured for the child. Again the same rule is valid, with an additional factor giving the concurrent number of forks. For example, 10 concurrent forks require:

```
(1 + 1 + 1) * 32bytes/page * 5MB * 256pages/MB * 10 forks = 1.2MB ECSA
```

Note that ptrace() also uses captured storage. Therefore, if a programmer debugs a 1MB program and 200KB of automatic stack data, you need:

```
(1 + 1 + 1) * 32bytes/page * 1.2MB * 256pages/MB = 29KB ECSA
```

Note: On a production system this might be sufficient, but if several programmers are debugging at the same time, you have to increase this value accordingly.

The use of memory by functions like mmap(), shmat(), and so on, can be controlled through settings in the UNIX Services member of PARMLIB. The actual usage of these settings can be monitored with RMF's OMVS Kernel Activity Report. We give an example of such a report in 4.5, "RMF OMVS Kernel Activity Report" on page 141.

3.6 Sizing for the OS/390 UNIX Services Environment

The typical question you will have to answer is, *"I have a UNIX application running on a machine having 1700 tpm TPC-C. What S/390 capacity do I have to allocate for it?"*

We see that the UNIX way of capacity planning is not a simple matter, mainly due to the fact that TPC-C scaling is not appropriate for S/390 because S/390 provides service for various kinds of loads, transactions (CICS, IMS), interactive systems (TSO/E), and batch on *one* system. This requires a certain investment in software to manage these types of loads simultaneously and assign business goals to them (see Figure 21 on page 106). So we have to look for a bridging model that gives us at least an *estimate* of a S/390 processor capable of running this load. The question is how to link these two worlds (see Figure 26).

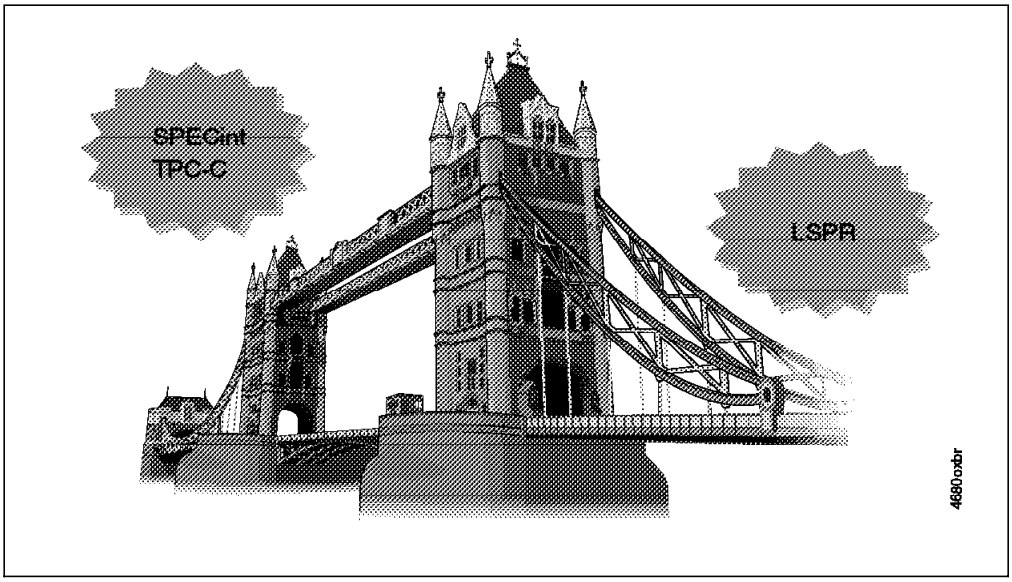


Figure 26. The Relative Capacity Problem

IBM is developing tools to address this issue. At the time of writing, tools have been tested in various server integration projects. Your IBM representative can find support on the IBM Intranet under:

```
http://w3.s390.ibm.com/s390ncc/nccinfo
```

3.6.1 The Bridge Model

Since the TPC-C benchmark has been accepted in the UNIX world as the measure of choice for commercial performance, it is necessary to use those results in generating an estimator of capacity.

1. We chose to use the CICS/DB2 environment, which is available on IBM RS/6000 machines and S/390 processors. Based on measurements in these environments, we found a RS/6000 59H processor to be equivalent to a S/390 CMOS 9672-R12 processor.
2. A RS/6000-59H is about 1340 tpm based on RS/6000 TPC-C measurements and its “relative commercial index.”
3. In case the application is really CPU-bound, the range may vary between 0.35 59H and 0.65 59H per 9672-R12.
4. If the application to be moved is very I/O intensive, and contentious OLTP type work, we assume 2.5 59H per 9672-R12.
5. You then have to figure out at what CPU utilization the UNIX systems run. In addition, if they have mixed applications on a machine, you have to check how much capacity is used by each.

Disclaimer

These are the major steps on which the transformation is based.

It should be emphasized that:

- We relate UNIX machines by their power in terms of tpm-C.
- We use the RS/6000-59H as a bridge to the S/390 world, based on IBM products (CICS/DB2).
- Due to the various assumptions and multipliers the tools use, they may be expected to lack some precision. This may change as we gain more experience in using them.

We show two examples of how to use a sizer tool in 7.4, “The OS/390 UNIX Services Sizer Tool” on page 240.

Chapter 4. Using RMF Parallel Sysplex Reports

At a Glance

This chapter gives an overview of the RMF components for historical reporting of Parallel Sysplex resources.

The RMF Coupling Facility Activity report is described in detail and it is shown how the data contained in the report can be used for capacity planning.

Recommended Reading:

- *OS/390 RMF Performance Management Guide*, SC28-1951
- *OS/390 RMF User's Guide*, SC28-1949
- *OS/390 RMF Report Analysis*, SC28-1950
- *CF Reporting Enhancements to RMF V5.1.*, WSC Flash 9609
- *OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook*, SG24-2076
- *OS/390 RMF Performance Management Guide*, SC28-1951
- *OS/390 RMF Report Analysis*, SC28-1950
- *OS/390 RMF User's Guide*, SC28-1949

© Copyright IBM Corp. 1998

4.1 RMF Parallel Sysplex Reports

In addition to CPU, storage, I/O and workload data, RMF collects information for the CF and XCF. The data is stored in SMF records type 74 subtype 4 and 2.

Figure 27 shows the RMF components that collect data and report on CF and XCF activity. The CF SMF data is collected and recorded by RMF Monitor III. RMF writes an SMF record for each CF per interval from each system in the Parallel Sysplex that is connected to the CF. The SMF data can be processed by the RMF Postprocessor, which combines the SMF records from all systems to create a sysplex-wide view of CF usage.

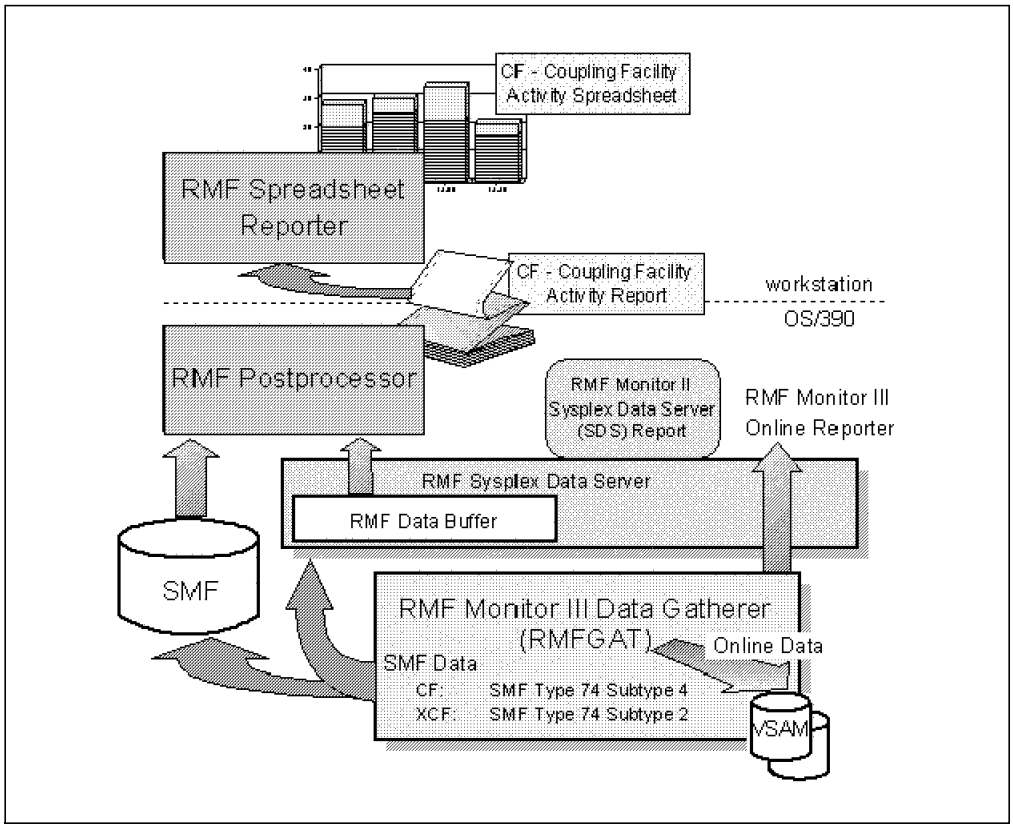


Figure 27. RMF Data Collection and Reporting for CF Resources

RMF Monitor III Data Gatherer

You must start the RMF Monitor III data gatherer (RMFGAT) on all systems from which you want to collect data for CF utilization. RMFGAT records various SMF records in addition to its online data. The SMF data is stored in the active SMF data set at the end of the RMF Monitor I data collection interval. To ensure that RMF Monitor III, RMF Monitor I, and other SMF data collection components are synchronized, it is recommended to use the default timing values as specified in the RMF parmlib members ERBRMF00 and ERBRMF04.

RMF Monitor III collects SMF data for the following resources:

- SMF Type 72 subtype 2, workload storage usage in compatibility mode
- SMF Type 72 subtype 4, workload storage usage in goal mode
- SMF Type 74 subtype 1, XCF utilization
- SMF Type 74 subtype 3, OS/390 UNIX Services
- SMF Type 74 subtype 4, CF utilization

RMF Sysplex Data Server

The RMF Sysplex Data Server provides a set of application programming interfaces that give application programs the capability to retrieve RMF data immediately after the data has been recorded. RMF Monitor II and RMF Monitor III online data can be accessed as soon as the data gatherer has been started on all systems in the Parallel Sysplex.

RMF Data Buffer

The RMF data buffer keeps RMF SMF data available in a data space on each system in the sysplex. Application programs, such as the RMF Postprocessor, can access and process the SMF data without using the SMF data set immediately after the SMF data has been recorded.

The data buffer is organized as a wrap-around buffer, which means that the oldest SMF records are replaced when the buffers overflow.

You must start data buffers on all systems to have access to all SMF data in the Parallel Sysplex. By default, the size of a buffer is 32MB and RMF SMF records type 70 to 78 will be copied to the buffer. The RMF Monitor II Sysplex Data Server report helps you to determine the wrap-around time of the buffer, the number of API calls issued to the data server, and their service times.

RMF Postprocessor

The RMF Postprocessor running in one image can process SMF data from the SMF data set as well as from the RMF data buffers. The RMF Postprocessor creates sysplex-wide reports for CF activity, shared DASD activity and workload activity (goal mode only). The XCF activity and the cache subsystem activity reports provide additional information for shared resources in a Parallel Sysplex environment.

The sysplex-wide reports are produced using the RMF Postprocessor control statement SYSRPTS. To generate the CF report, specify: `SYSRPTS(CF)`.

RMF Spreadsheet Reporter

The RMF Spreadsheet Reporter is an extension to the RMF Postprocessor running on a workstation to graphically display RMF report data with spreadsheet applications (for more details, refer to A.3, “RMF Spreadsheet Reporter” on page 259). The Spreadsheet Reporter contains a spreadsheet macro for the RMF CF report (see 4.2.6, “RMF CF Activity Spreadsheet” on page 133).

4.2 Coupling Facility Reports

The data for this report is gathered individually by each instance of the Monitor III data gatherer in the Parallel Sysplex and stored in one SMF record (74.4) per Coupling Facility per interval. A Postprocessor running in one image accesses this data through Parallel Sysplex Data Server services. The report is produced for each Coupling Facility in the Parallel Sysplex and has three sections:

- CF Activity - Usage Summary (see Figure 28 on page 123).
- CF Activity - Structure Activity (see Figure 29 on page 124).
- CF Activity - Subchannel Activity (see Figure 30 on page 126).

This set of reports assists the installation in managing the Coupling Facilities. They are helpful for:

- Optimizing structure placement

- Tuning the structure allocation policy in the couple data set
- Doing Parallel Sysplex capacity planning by workload type

In the discussions that follow, references are made to sample reports. For example, the notation **A** in the following text corresponds to the notation **A** in the Coupling Facility reports presented in Figure 28 on page 123.

4.2.1 RMF CF Activity - Usage Summary

This section summarizes all activities for the CF. It is divided into three parts that show the structure **A**, storage **B**, and processor **C** utilization of the CF. You can use this section to evaluate the status of all structures and how the CF as a whole is being utilized.

4.2.1.1 Structure Summary

The structure summary lists all structures ordered by type **1** and name **2**. For each structure the status **3** at the end of the RMF data collection interval is presented:

- ACTIVE: Storage is allocated and there is connection to at least one system.
- INACTV: Storage is allocated but there is no connection to any system. Persistent structures, like IRLM lock structures, will remain in the CF even if no system is connected to the structure anymore.
- UNALLOC: The structure was active at some point in time during the interval and did not exist at the end of the RMF interval.

For each active structure, the section shows the allocated structure size **4**, the total number of requests **5**, the average number of requests/second, and the number of allocated **6** and used directory and data elements. For cache structures, the number of directory reclaims **7** is shown in the last column.

The information presented in this part of the report can be used to validate the size of various types of structures. In most cases, some knowledge of the structure is required to interpret the data.

Cache structures

The TOT/CUR (total/current) in-use values indicate if the structure is too big (if it never fills up) or misproportioned in terms of entry-to-element ratio (one never fills up but the other does). These values will not help in knowing whether the structure is too small (that is, 100% full may be normal).

The DIR RECLAIMS (directory reclaims) can be used as an indicator whether the directory of a cache structure is over-committed. Whenever a shortage of directory entries occurs, the CF reclaims in-use directory entries associated with unchanged data. These reclaimed items are used to satisfy new requests for directory entries by the data base manager. All users of the data item represented by the directory entry must be notified that their copy of the data item is invalid. When the data base manager needs access to the now invalidated data item, the item must be reread from DASD and registered with the CF.

Avoid Directory Reclaims

Directory reclaim activity results in increased I/O activity to the data base to reacquire a referenced data item. To reacquire data items and to re-register them to the CF may result in increased CPU utilization and elongated transaction response times whenever a read miss occurs in the local buffer pool.

Directory reclaims can be eliminated by increasing the number of directory entries for a particular structure. This can be accomplished by:

- Increasing the size of the structure

 For directory-only structures, only the number of directory entries is affected. For structures with data elements and directory entries, both will increase in the ratio specified.

- Changing the proportion of the structure

 Some cache structure exploiters allow the installation to specify the ratio of directory elements to data elements. DB2 and IMS provide this capability. RACF, for example, is hard-coded and uses a 1:1 ratio of directory entries to data elements.

 It is difficult to estimate the impact of changing the directory entry to data element ratio unless there is a consistent difference between the TOT and CUR values. The CUR value reported is not a highwater mark or average value, it is the number in use at the end of the interval.

List structures

Some list structures, like the JES2 checkpoint structure, are very static in size. For these structures, the TOT can be only slightly larger than CUR. Other functions, like the system logger, off-load data as the structure fills up. The threshold that triggers the off-load is based on parameters associated with the function. The ratio of CUR to TOT should not be higher than this threshold. Other list structures, such as the XCF structure and the VTAM Generic Resource structure, must handle large peaks of data. For these structures, the TOT should be much larger than the CUR in-use to prevent backups of signals during spikes.

Lock structures

Lock structures, such as IRLM lock structures, are divided into two parts:

1. The actual lock table contains the number of entries in the LOCK ENTRIES TOT column. The number of LOCK ENTRIES CUR is a sampled value of the number in use at the end of the interval. The best way to evaluate the size of this table is to look at the FALSE CONTENTION **13** data for the lock structure on the Coupling Facility Structure Activity report. For a discussion on false contention refer to “False Contention” on page 54.
2. The second part of the lock structure contains record data entries. The number of these entries is reported in the LIST ENTRIES column. If there are not enough of these entries, you will not be able to obtain locks, and transactions will start failing at that point.

IRLM provides no external means of apportioning the structure storage between record data and locks; you can only obtain adequate entries for both parts by judiciously choosing the size of the lock structure. The easiest way to do this is to use the structure size tables in *OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook*, SG24-2076.

4.2.1.2 Storage Summary

The storage summary **B** section reports the total amount and percentages of storage that is occupied with structures. Ensure that sufficient space is available to allow for rebuild from another CF.

4.2.1.3 Processor Summary

The processor summary **C** section shows the CF model and version **8** and indicates the CP utilization **9** of the LP that hosts the CF. LOGICAL PROCESSORS: DEFINED **10** shows how many logical processors are defined to this CF. The EFFECTIVE number reflects how many of those defined engines the CF used, including the time spent looking for work (only of interest if you use PR/SM).

```
                                C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                                                PAGE    1
     OS/390                  SYSPLEX PRODPLEX           DATE 07/08/1997          INTERVAL 015.00.000
     REL. 02.04.00            RPT VERSION 2.4.0          TIME 10.00.00            CYCLE 01.000 SECONDS

------------------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = CF01
 TOTAL SAMPLES(AVG) =   896  (MAX) =   898  (MIN) =   894
------------------------------------------------------------------------------------------------------------------------------------
                                              COUPLING  FACILITY  USAGE  SUMMARY
------------------------------------------------------------------------------------------------------------------------------------
 STRUCTURE SUMMARY  A
------------------------------------------------------------------------------------------------------------------------------------
 1       2                    3              4                    5                                    6                          7
                                                    % OF                      % OF      AVG     LST/DIR    DATA       LOCK      DIR
         STRUCTURE                         ALLOC    CF             #         ALL       REQ/    ENTRIES    ELEMENTS   ENTRIES   RECLAIMS
 TYPE    NAME                 STATUS CHG   SIZE     STORAGE        REQ       REQ       SEC     TOT/CUR    TOT/CUR    TOT/CUR

 LIST    IXCLIST1             ACTIVE       20M        1.0%       1345K     44.4%     1494.9    4769       4755       N/A       N/A
                                                                                                  19         38       N/A
         IXCLIST2             ACTIVE       20M        1.0%       553882    18.3%     615.42    4769       4755       N/A       N/A
                                                                                                  26         45       N/A
         LOGS001              ACTIVE        2M        0.1%          512     0.0%       0.57    1978       3958       N/A       N/A
                                                                                                  12         98       N/A
         LOGS002              ACTIVE        1M        0.1%            0     0.0%       0.00     950       1899       N/A       N/A
                                                                                                  11         96       N/A
         OPERLOG              ACTIVE       20M        1.0%         5171     0.2%       5.75     21K         63K      N/A       N/A
                                                                                                4026       8031       N/A

 LOCK    ISGLOCK              ACTIVE       20M        1.0%       288887     9.5%     320.99       0          0       2097K     N/A
                                                                                                   0          0         31K
         MMDB2_LOCK1          ACTIVE      100M        5.0%       758529    25.0%     842.81     269K         0         34M     N/A
                                                                                                4062          0         49K
         NNDB2_LOCK1          ACTIVE       50M        2.5%        11007     0.4%      12.23     135K         0       8389K     N/A
                                                                                                 231          0         12K
         NNIMSIRLM            ACTIVE       10M        0.5%          163     0.0%       0.18      45K         0       1049K     N/A
                                                                                                   0          0         21

 CACHE   IMSVSAM              ACTIVE       75M        3.8%        29097     1.0%      32.33     401K         0       N/A         0
                                                                                                  53K         0       N/A
         IRRXCF               ACTIVE       20M        1.0%        32183     1.1%      35.76    4863       4851       N/A         0
                                                                                                4847       4847       N/A
         MMDB2_GBP            ACTIVE       40M        2.0%         8297     0.3%       9.22     41K       8215       N/A         0
                                                                                                6236       3339       N/A
                                          -------      -----     -------    ------    -------
             STRUCTURE TOTALS             378M       19.0%      3033K      100%     3370.2

 STORAGE SUMMARY  B
------------------------------------------------------------------------------------------------------------------------------------

                                                 ALLOC      % OF CF      ------- DUMP SPACE -------
                                                 SIZE       STORAGE      % IN USE   MAX % REQUESTED

 TOTAL CF STORAGE USED BY STRUCTURES              378M       19.0%
 TOTAL CF DUMP STORAGE                             20M        1.0%          0.0%              0.0%
 TOTAL CF STORAGE AVAILABLE                         2G       80.0%
                                                 -------
 TOTAL CF STORAGE SIZE                              2G

                                                 ALLOC      % ALLOCATED
                                                 SIZE

 TOTAL CONTROL STORAGE DEFINED                      2G       20.0%
 TOTAL DATA STORAGE DEFINED                        0K        0.0%

 PROCESSOR SUMMARY  C
------------------------------------------------------------------------------------------------------------------------------------

 8  COUPLING FACILITY MODEL   9674      VERSION C04      CFLEVEL 3
 9  AVERAGE CF UTILIZATION  (% BUSY)          40.7      LOGICAL PROCESSORS:   DEFINED   1    EFFECTIVE   1.0  10
```

Figure 28. RMF CF Activity Report - Usage Summary

4.2.2 RMF CF Activity - Structure Activity

This section of the report shows in detail the activity for each structure per connected system. The information presented can be used to identify causes for large response times.

```
                                C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                                    PAGE   7
     OS/390                  SYSPLEX PRODPLEX           DATE 07/08/1997          INTERVAL 015.00.000
     REL. 02.04.00            RPT VERSION 2.4.0          TIME 10.00.00            CYCLE 01.000 SECONDS

 ------------------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = CF01
 ------------------------------------------------------------------------------------------------------------------------------------
                                          COUPLING  FACILITY  STRUCTURE  ACTIVITY
 ------------------------------------------------------------------------------------------------------------------------------------

  STRUCTURE NAME = OPERLOG           TYPE = LIST  D
            # REQ    -------------- REQUESTS --------------   --------------- DELAYED REQUESTS -------------
  SYSTEM    TOTAL             #     % OF  -SERV TIME(MIC)-    REASON    #   % OF  ---- AVG TIME(MIC) -----
  NAME      AVG/SEC           REQ    ALL     AVG   STD_DEV            REQ    REQ   /DEL    STD_DEV   /ALL

  J80         918    SYNC    687   13.3%   425.6    208.7
             1.02    ASYNC   224    4.3%  2112.8   1749.1     NO SCH   10   4.3%  819.4       1170   35.5
                     CHNGD     7    0.1%  INCLUDED IN ASYNC
                                                              DUMP       0   0.0%    0.0        0.0

...                                  11                                  12

 ------------------------------------------------------------------------------------------------------------------------------------
  TOTAL      5171    SYNC   4899   94.7%   341.9    575.2
             5.75    ASYNC   230    4.4%  2229.7   2635.5     NO SCH   50  18.4%   5718      21716   1051
                     CHNGD    42    0.8%
                                                              DUMP       0   0.0%    0.0        0.0    0.0


  STRUCTURE NAME = MMDB2_LOCK1       TYPE = LOCK  E
            # REQ    -------------- REQUESTS --------------   --------------- DELAYED REQUESTS -------------        13
  SYSTEM    TOTAL             #     % OF  -SERV TIME(MIC)-    REASON    #   % OF  ---- AVG TIME(MIC) -----  EXTERNAL REQUEST
  NAME      AVG/SEC           REQ    ALL     AVG   STD_DEV            REQ    REQ   /DEL    STD_DEV   /ALL   CONTENTIONS

  J80         194K   SYNC    194K  25.6%   162.2    303.2                                                    REQ TOTAL      327K
             215.7   ASYNC     0    0.0%     0.0      0.0     NO SCH    0   0.0%    0.0        0.0    0.0   REQ DEFERRED    40K
                     CHNGD     0    0.0%  INCLUDED IN ASYNC                                                 -CONT           40K
                                                                                                            -FALSE CONT   4605

...
 ------------------------------------------------------------------------------------------------------------------------------------
  TOTAL       759K   SYNC    759K   100%   163.3    337.0                                                    REQ TOTAL     1258K
             842.8   ASYNC     0    0.0%     0.0      0.0     NO SCH    0   0.0%    0.0        0.0    0.0   REQ DEFERRED   124K
                     CHNGD     0    0.0%                                                                    -CONT          124K
                                                                                                            -FALSE CONT     17K


  STRUCTURE NAME = MMDB2_GBP         TYPE = CACHE  F
            # REQ    -------------- REQUESTS --------------   --------------- DELAYED REQUESTS -------------
  SYSTEM    TOTAL             #     % OF  -SERV TIME(MIC)-    REASON    #   % OF  ---- AVG TIME(MIC) -----
  NAME      AVG/SEC           REQ    ALL     AVG   STD_DEV            REQ    REQ   /DEL    STD_DEV   /ALL

  J80        2047    SYNC   2012   24.2%   204.9    122.9
             2.27    ASYNC    23    0.3%  6446.4   5176.7     NO SCH   19  54.3%  501.3      380.5  272.1
                     CHNGD    12    0.1%  INCLUDED IN ASYNC
                                                              DUMP       0   0.0%    0.0        0.0

...
                                                                                                                    14
 ------------------------------------------------------------------------------------------------------------------------------------
   TOTAL     8297    SYNC   7991   96.3%   225.3    395.5                                                    -- DATA ACCESS ---
             9.22    ASYNC   254    3.1%  5661.8   5599.3     NO SCH  137  44.8%  1050       2806  470.1   READS         1203
                     CHNGD    52    0.6%                                                                    WRITES        1573
                                                              DUMP       0   0.0%    0.0        0.0    0.0   CASTOUTS       842
                                                                                                            XI'S          9204
```

Figure 29. RMF CF Activity Report - Structure Activity

The structure activity report presents, for all structures, the number of synchronous and asynchronous requests (section REQUESTS **11** in Figure 29)

and the average service times and standard deviation for these requests (SERV TIME(MIC)). The DELAYED REQUESTS **12** fields show the number of asynchronous requests that were delayed due to subchannel busy conditions. The /DEL field indicates the average delay time for each delayed request, and /ALL the amortized delay time. 4.2.4, “Understanding CF Processing” on page 126 discusses potential delay reasons for CF requests, and how they are presented in the RMF CF Activity Report, in more detail.

The structure activity section also gives information on lock contention for serialized list and lock structures **13**, as well as data access statistics for cache structures **14**. A discussion for different structures follows:

Serialized List Structures

A serialized list structure contains an associated lock structure. For example, JES2 and VTAM structures are serialized list structures and the XCF structure is not a serialized list structure. A serialized list structure can provide the structure user with the possibility of exclusive usage over the entire structure, a portion of the entries, or a single entry.

RMF reports on the number of accesses to the serialized list structure and the number of accesses deferred due to lock contention. A serialized list structure will not incur false contention due to the way in which XES manages the contention.

Lock Structures

The lock structure exploiter completely controls access to the lock structure. RMF reports on the number of requests issued to the XES API for the lock structure (REQ TOTAL **13**), and the number of requests deferred due to contention (REQ DEFERRED).

The number of external requests can be significantly higher than the number of total requests (# REQ – TOTAL) issued to the CF. This is because XES optimizes the number of unlock requests issued to the structure (see **13**).

A request can be deferred due to true lock contention, false lock contention, and XES internal processing delays. Refer to 2.1.3, “Locking in Parallel Sysplex” on page 53 for a discussion about real and false contention.

As a result of lock contention you may see increased XCF activity caused by the exploiter of the lock structure negotiating the lock status.

The lock structure **E** in Figure 29 on page 124 shows high values for true lock contention **13** (more than 10%). The false lock contention is a little bit more than 1% of the deferred requests, which is still in an acceptable range. The true contention should be further investigated. True contention is application dependent; therefore, it is necessary to identify the workload using the lock structure. For example, long-running batch jobs that do not commit the locks can cause high true contention on lock structures.

Cache Structures

Data access statistics **14** for cache structures **F** are acquired from counters in the CF; they cannot be broken down into individual connection contributions. Only the owner (database manager) of the cache structure knows how efficiently the local buffer and cache structure are being used. Depending on the cache structure implementation, one or more counts can be zero.

- READS and WRITES indicate how often the CF returned data to a connector and how often a connector placed changed data into the cache structure. If you

observe a high number of WRITES and a small number of READS combined with large service times, you may want to check the ratio of directory entries to data elements. If directory reclaims **7** are high, too, you may want to increase the structure size.

- CASTOUTS is a count of the number of times a connector retrieved a changed data entry, wrote the data to DASD and caused the changed attribute to be reset to unchanged. This counter is of interest for store-in cache structures (for example, DB2 group buffer pools) and can be ignored for store-through cache structures (for example, CICS/VSAM RLS structures).
- XIs shows the number of times a data item residing in a local buffer pool was marked invalid by the CF during the RMF interval. XIs is used for directory-only, store-in and store-through cache structures. The counter reflects the amount of data sharing among the users of the cache structure and the amount of write/update activity against the data bases.

4.2.3 RMF CF Activity - Subchannel Activity

This section of the report summarizes the activity per system to the CF. The information in this section is similar to that in the Structure Activity section. In addition, you can examine the path **15** and subchannel **16** busy counts for all requests. The content of this section is discussed in detail in 4.2.4, “Understanding CF Processing.”

```
                                C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                                                                                                PAGE  15
     OS/390                   SYSPLEX PRODPLEX            DATE 07/08/1997           INTERVAL 015.00.000
     REL. 02.04.00             RPT VERSION 2.4.0           TIME 10.00.00             CYCLE 01.000 SECONDS

 ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
  COUPLING FACILITY NAME = CF01
 ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
                                                                SUBCHANNEL  ACTIVITY
 ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
           # REQ                                  ----------- REQUESTS -----------   -------------- DELAYED REQUESTS --------------
  SYSTEM   TOTAL                  --BUSY--              #   -SERVICE TIME(MIC)-          #     % OF  ------ AVG TIME(MIC) ------
  NAME     AVG/SEC -- CONFIG --  -COUNTS-              REQ     AVG    STD_DEV           REQ    REQ   /DEL    STD_DEV     /ALL
                                   15                                                    16
  J80      641905  SCH GEN    4  PTH   23K    SYNC    220523   163.7     297.7    SYNC  4069   1.8%  371.1       1656      6.8
            713.2  SCH USE    4  SCH 4069     ASYNC   405784   837.7      1500    ASYNC   19K  4.7%   1013       3557     47.1
                   SCH MAX    4               CHANGED    164  INCLUDED IN ASYNC  TOTAL   23K  3.7%
                   PTH        2               UNSUCC       0     0.0       0.0
...
  K11      256184  SCH GEN    4  PTH 9035     SYNC     88681   171.5     433.4    SYNC   138   0.2%  914.7       2813      1.4
            284.6  SCH USE    4  SCH  138     ASYNC   164490  1069.9      1011    ASYNC 3280   2.0%  661.1       1495     13.2
                   SCH MAX    4               CHANGED     13  INCLUDED IN ASYNC  TOTAL 3418   1.4%
                   PTH        2               UNSUCC       0     0.0       0.0
  K12      440421  SCH GEN    4  PTH   10K    SYNC    240203   144.5     299.3    SYNC   727   0.3%  246.4      977.8      0.7
            489.4  SCH USE    4  SCH  727     ASYNC   194720   909.5      1255    ASYNC 5492   2.8%   1153       3034     32.5
                   SCH MAX    4               CHANGED     46  INCLUDED IN ASYNC  TOTAL 6219   1.4%
                   PTH        2               UNSUCC       0     0.0       0.0
```

Figure 30. RMF CF Activity Report - Subchannel Activity

4.2.4 Understanding CF Processing

Chapter 5, “A CF Processing Model” on page 143 describes how a request flows from the CEC to the CF and back. A CF request is first processed in OS/390, then transferred to the CF if there is a free link, processed by a CF CP, and sent back to OS/390 via the CF link. This section discusses the possible delays for requests to the CF and how they are reflected in the RMF CF Activity Report.

Figure 31 on page 127 shows six possible situations of how synchronous (sync) and asynchronous (async) requests are processed if a subchannel or path is not

available. Each system has two subchannels to each link. The link has two buffers, one for each subchannel. A picture of a possible Parallel Sysplex configuration is shown in Figure 43 on page 145.

In the discussion that follows, references are made to the sample reports and the processing flow. The numbers (for example 1) in the following text refer to Figure 31, and lower case letters (for example a) to Figure 32 on page 129.

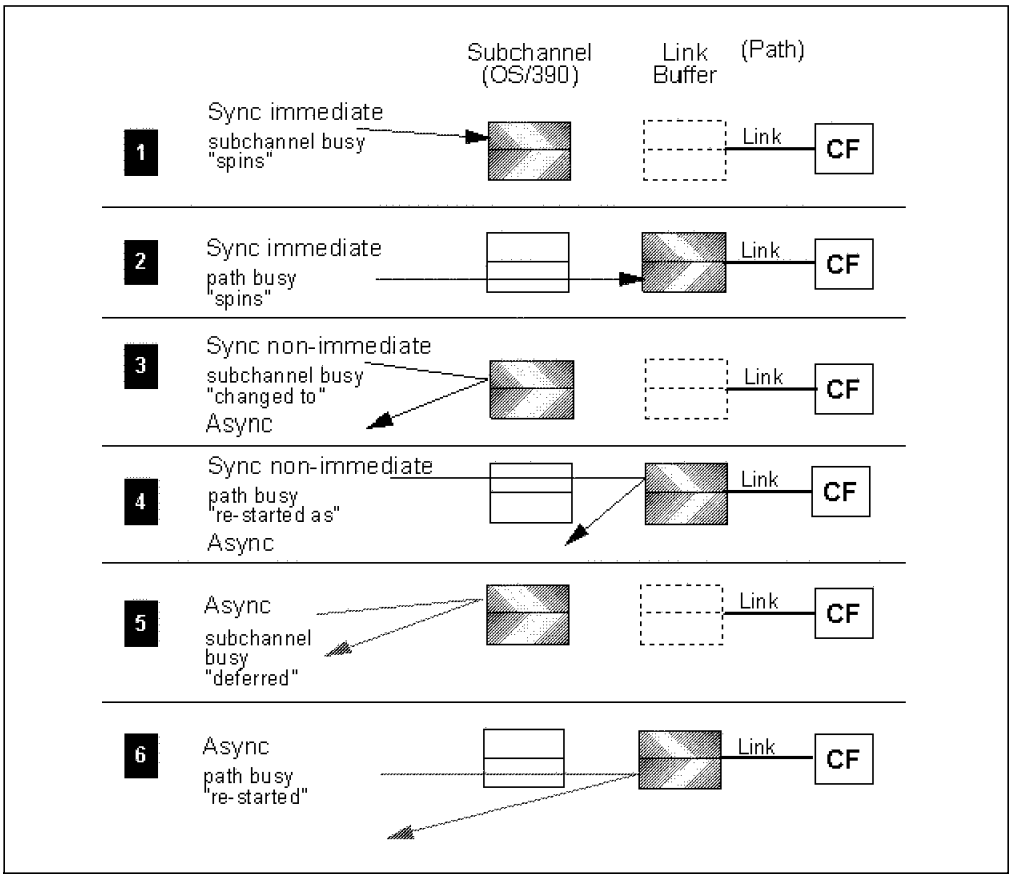


Figure 31. CF Requests and Delay Types

1 A synchronous, immediate request waits (“spins”) until the subchannel becomes available. The number of synchronous requests that were delayed because a subchannel was not available is reported in the Subchannel Activity section as DELAYED REQUESTS - SYNC c. The count is not available on a per structure basis; however, you may estimate the number for a particular structure from the Subchannel Activity section.

2 In this example, the synchronous immediate request gets a free subchannel but encounters a path busy condition. This situation can occur in a PR/SM environment where multiple OS/390 systems share the same link to the CF. The request “spins” until the link becomes available.

RMF reports BUSY COUNTS - PTH d in the subchannel activity section. The time the synchronous request was delayed due to path busy is part of the synchronous service time. Therefore, you may want to check the path busy count if you see long synchronous service times in a PR/SM environment with EMIF.

3 A synchronous, non-immediate request that encounters a subchannel busy condition is changed to an asynchronous request. RMF reports the changed requests in CHNGD **a** for each structure in the Structure Activity section and in CHANGED **b** for each system in the Subchannel Activity section. From now on the request is asynchronous and is processed as described in case **5**.

4 A synchronous, non-immediate request that obtains a free subchannel but encounters a path busy condition is restarted in OS/390 as an asynchronous request. The path busy count **d** on the Subchannel Activity section is updated. The changed indicators **a** and **b** are not updated. This situation is identical as for asynchronous requests described in case **6**.

5 An asynchronous request that is delayed because the subchannel was not available is queued to the subchannel queue in OS/390. The number of asynchronous requests delayed and the delay time for those requests is reported in the Structure Activity section for each structure as DELAYED REQUESTS - NO SCH **e**. The DELAYED REQUESTS - ASYNC **f** in the Subchannel Activity section summarizes the number of delayed requests for all structure requests from this system. The time an asynchronous request was delayed is shown in DELAYED REQUESTS - /DEL. This time is not accumulated in the service time for asynchronous requests.

6 In a PR/SM environment, an asynchronous request that encounters a path busy condition is re-queued in OS/390. In this case, the operation starts again, which means the request is counted as a new request on the Subchannel Activity section (counted in # REQ **h**). The busy condition is also added to BUSY COUNTS - PTH **d** in the Subchannel Activity section.

The # REQ **p** on the Structure Activity section is not increased in this case. This value is the sum of the SYNC, ASYNC, and CHNGD counts on the Structure Activity section. The value shown in # REQ on the Subchannel Activity section can be higher because of path busy conditions for asynchronous requests.

The total number of requests on the Subchannel Activity section # REQ - TOTAL **h** is a count that shows all requests started to the CF. The sum of SYNC **q**, ASYNC **i**, and CHANGED **b** requests on the Subchannel Activity section is the number of requests that were issued by OS/390 applications and completed. Subtraction of the sum of **q**, **i**, and **b** from the total **h** yields a number which includes asynchronous requests which encountered a path busy condition but also internal XES requests. The latter number is usually small; therefore, in a PR/SM environment, most of the requests counted in **h** are due to path busy conditions for asynchronous requests.

Using this calculation for system K11 in the Subchannel Activity section in Figure 32 on page 129 shows that 3000 requests were issued to the CF which are not accounted for in the sum of synchronous, asynchronous and changed requests. We know that some portion of it is due to internal XES requests, but most of it is because of path busy conditions for asynchronous requests. If we subtract this value from the path busy count **d**, we see that about 3000 asynchronous requests and about 6000 synchronous requests were delayed because of path busy.

```
                                        COUPLING  FACILITY  STRUCTURE  ACTIVITY

          # REQ    -------------- REQUESTS -------------   -------------- DELAYED REQUESTS -------------
SYSTEM    TOTAL             #     % OF  -SERV TIME(MIC)-    REASON    #   % OF  ---- AVG TIME(MIC) -----
NAME      AVG/SEC         REQ     ALL     AVG   STD_DEV             REQ    REQ   /DEL    STD_DEV   /ALL

            p
K11         114    SYNC    112    2.2%   316.4    122.6
            0.13   ASYNC  g  1    0.0%  1179.0    288.5    NO SCH e  2   100%  10355      13373  10355
                   CHNGD  a  1    0.0%  INCLUDED IN ASYNC
                                                 DUMP       0   0.0%    0.0        0.0

K12        3543    SYNC   3515   68.0%   321.4    603.6
           3.94    ASYNC     2    0.0%  3661.0   6356.5    NO SCH   30   107% k 3037       4331   3254
                   CHNGD    26    0.5%  INCLUDED IN ASYNC
                    l                                    DUMP       0   0.0%    0.0        0.0


                                              SUBCHANNEL  ACTIVITY

          # REQ                              ----------- REQUESTS -----------   -------------- DELAYED REQUESTS --------------
SYSTEM    TOTAL                  --BUSY--             #   -SERVICE TIME(MIC)-          #     % OF  ------ AVG TIME(MIC) ------
NAME      AVG/SEC -- CONFIG --   -COUNTS-             REQ     AVG    STD_DEV          REQ    REQ   /DEL    STD_DEV     /ALL
            h       m             d                   q                    c
K11       256184  SCH GEN    4  PTH 9035    SYNC     88681   171.5     433.4   SYNC   138   0.2%  914.7       2813      1.4
           284.6  SCH USE    4  SCH  138     ASYNC  i 164490  1069.9      1011   ASYNC 3280  f 2.0%  661.1       1495     13.2
                  SCH MAX    4   j           CHANGED  b  13  INCLUDED IN ASYNC  TOTAL 3418   1.4%
                  PTH        2              UNSUCC       0     0.0       0.0     n
            o
K12       440421  SCH GEN    4  PTH   10K    SYNC    240203   144.5     299.3   SYNC   727   0.3%  246.4       977.8      0.7
           489.4  SCH USE    4  SCH  727     ASYNC   194720   909.5      1255   ASYNC 5492   2.8%   1153       3034     32.5
                  SCH MAX    4              CHANGED      46  INCLUDED IN ASYNC  TOTAL 6219   1.4%
                  PTH        2              UNSUCC       0     0.0       0.0
```

Figure 32. RMF CF Activity - Structure and Subchannel Activity Sections

Table 11 on page 130[10] summarizes the delay reasons and how the counters in the RMF CF Activity report are updated. The references **1**, **2**, ..., **6** refer to the discussed delay types in Figure 31 on page 127. For each discussed delay reason the table shows:

- Total number of requests as shown on the subchannel and structure activity report.
- Request counts in the REQUESTS sections and time values in the SERVICE TIME sections of the RMF CF Activity report. The request counts and the average service times are updated when the requests complete.
- Delay counts and average delay times. The delay counter is updated when the delay is encountered. The delay time is updated when the requests can be sent to the CF.
- Path and subchannel busy counts as shown on the subchannel activity report.

For all counts it is shown by how much they are increased in case of the event. For all time values a checkmark (√) indicates whether the value is being updated.

[10] The table reflects how the counters are updated when a busy condition occurs before the request completes. It does not reflect the counters when no busy condition occurs.

Table 11. Summary of Delay Reasons and Affected Counters in RMF CF Activity Report

Report fields	Sync immediate		Sync non-immediate		Async	
	1	2	3	4	5	6
Total requests on subchannel (1) and structure (2) activity section						
• # REQ (1)	+ 1	+ 1	+ 1	+ 2 (a)	+ 1	+ 2 (a)
• # REQ (2)	+ 1	+ 1	+ 1	+ 1	+ 1	+ 1
The following report fields are updated when the delay occurs						
DELAYED REQUESTS sections						
• SYNC (c)	+ 1					
• NOSCH/ASYNC (d)			+ 1		+ 1	
DELAYED REQUESTS – AVG TIME						
• SYNC	√					
• ASYNC			√		√	
BUSY COUNTS (Subchannel Activity section)						
• PATH		+ 1		+ 1		+ 1
• SCH	+ 1					
The following counts are updated when the delay is no longer present and the request is sent to the CF						
REQUESTS sections						
• SYNC	+ 1	+ 1				
• ASYNC				+ 1	+ 1	+ 1
• CHNGD (b)			+ 1			
REQUESTS – SERVICE TIME (e) sections						
• SYNC	√	√				
• ASYNC			√	√	√	√

Note:

(1) the count is updated when the request is started to the CF

(2) this is the sum of SYNC, ASYNC, and CHNGD fields on the Structure Activity section.

(a) this request is counted twice, because it is restarted as a new request

(b) the field is named CHANGED on the Subchannel Activity section

(c) only on Subchannel Activity section

(d) named NOSCH on Structure Activity section and ASYNC on Subchannel Activity section

(e) named SERV TIME on Structure Activity section

Subchannel Busy

For subchannel busy you should use the counts and times shown in the DELAYED REQUESTS parts of the Structure and Subchannel Activity sections. The value BUSY COUNTS - SCH **j** is always the same as the number of DELAYED REQUESTS - SYNC **c**.

You may find that the field % OF REQ - NO SCH **k** exceeds 100%, as shown for system K12 in Figure 32 on page 129. In this case 30 retries were necessary to process the 28 asynchronous requests (sum of ASYNC and CHNGD **l**).

In a non-PR/SM environment, subchannel busy is the indicator for path or link contention. An acceptable number for subchannel busy is less than 10% of all requests to the CF. If the value exceeds this value, you should ensure that the correct number of subchannels is defined in the I/O gen (see CONFIG **m**). If you conclude that you have a capacity problem, adding more paths may lessen or solve the problem.

Path Busy

In a PR/SM environment, you should check the BUSY COUNTS - PTH **d** as an indicator for path contention. A path busy count means the contention is due to activity from another LP. In this example, systems K11 and K12 are defined in the same CEC, and K12 sends about twice as many requests **o** to the CF as K11.

If this becomes critical, or if K11 is a production system, it is recommended to use dedicated paths to the CF. An acceptable value for path and subchannel busy is less than 10% of the total requests issued to the CF.

4.2.5 Summary of How to Read the RMF CF Activity Report

This section summarizes the descriptions and recommendations of the previous sections for the RMF CF Activity report. A recommendation on how to read through the data is included.

Start with Service Times

The most important information is the service time for each structure in the CF. You may want to start reading the report by examining the service times for all structures in the Structure Activity section.

The ITSO redbook *S/390 MVS Parallel Sysplex Performance*, SG24-4356, describes measured numbers from a series of 100% DB2, IMS DB and CICS VSAM/RLS data sharing benchmarks.

Measurements of CF lock requests are shown in Table 12 on page 132 for various combinations of sender CECs and CFs. This is the average service time per request, as shown in the RMF CF activity reports. Note that the IBM 9674s had a reasonable load on them, so that lower service times could be seen on lightly loaded CFs. The reported time includes data transfer time on the CF links, but not the time to set up the request.

Table 12. Coupling Facility Access Service Times for Lock Requests		
CEC	**Coupling Facility**	**Service Time**
9672 G4	9674-C05	0.07 ms
9672 G4	9674-C04	0.10 ms
9672 G3 *	9674-C04	0.14 ms
9672 R2/R3-based	9674-C02/3	0.18 ms
9672 R1-based	9674-C01	0.25 ms
9021 711-based	9674-C01	0.16 ms
9021 711-based	9674-C02/3	0.13 ms
9021 711-based	9674-C04	0.10 ms
9021 711-based	9021 711-based	0.08 ms
Note: Single-mode CF links are used to connect CEC and CF; CP speed for G3/G4 CECs and corresponding C04/C05 CFs vary slightly depending on the model.		
Note: * No HiPerLinks.		
Note: CF Utilization below 25%, without significant queueing.		

For cache requests sending 4KB blocks, approximately 0.04 ms must be added to the service time numbers in this table.

For asynchronous requests, the service times should be between 1 and 5 ms. The lower value refers to requests without data transfer (for example, lock requests) and is applicable only for the most recent hardware. The higher value refers to requests with data transfer (for example, DB2 sending one or more pages to the CF).

If all service times are within the acceptable range, you can stop examining the report.

Check CF Utilization

CF utilization and available CF storage helps you to understand how much capacity is available for tuning activities. As a conservative number, CF utilization should be no more than 40%. In some installations you may find that higher utilization of the CF still provides acceptable service times. This is especially the case for CFs with more than one CP.

You should ensure that sufficient storage is available to allow for structure rebuild from another CF.

Check Path and Subchannel Busy

For shared CF links, you should use the path busy counter; otherwise, the subchannel busy values of the DELAYED REQUESTS sections. In any case, the sum of path and subchannel busy counts should be less than 10% of the number of total requests to the CF.

Details for Cache Structures

For cache structures, the number of directory reclaims and eventually the read/write ratio can help you to adjust the size of the structure or to identify a performance problem (see 4.2.1, “RMF CF Activity - Usage Summary” on page 120 and 4.2.2, “RMF CF Activity - Structure Activity” on page 124).

Details for Lock Structures

For lock structures, the best way to evaluate the size of the structure is to look at the FALSE CONTENTION data. A discussion on false contention and guidelines for acceptable indicators are provided in “False Contention” on page 54.

Real contention cannot be resolved by simply manipulating the structure size. It depends very much on the characteristics of the workload exploiting the structures.

Details for List Structures

The Usage Summary section contains values for the number of list elements in use at the end of the interval. In some cases this can be helpful in adjusting the size of the structure (see discussion in 4.2.1, “RMF CF Activity - Usage Summary” on page 120).

4.2.6 RMF CF Activity Spreadsheet

The RMF Spreadsheet Reporter contains a spreadsheet macro that assists you in analyzing the RMF CF activity report. The RMF Spreadsheet Reporter is briefly discussed in A.3, “RMF Spreadsheet Reporter” on page 259. The *OS/390 RMF User's Guide*, SC28-1949, contains a complete description of this RMF function.

The idea of the macro is to combine multiple RMF CF activity reports in one spreadsheet to analyze one interval and trends across multiple intervals. Figure 33 illustrates the capabilities of the spreadsheet.

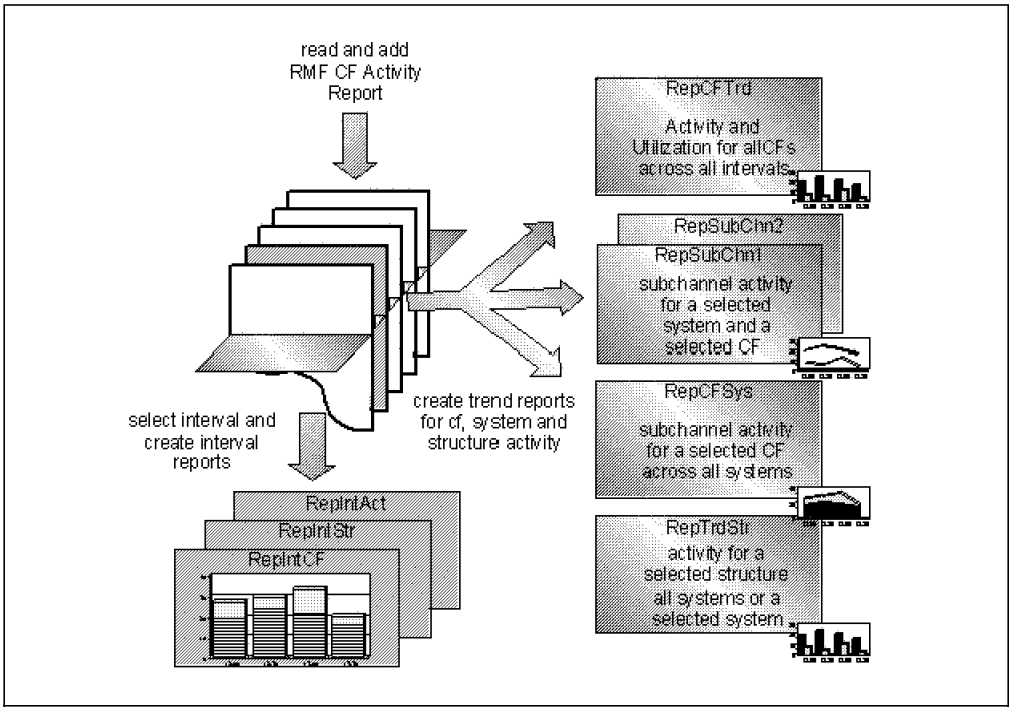


Figure 33. RMF CF Activity Spreadsheet - Structure

Figure 34 on page 134 shows a sample spreadsheet graphic for DB2 buffer pool MMDB2_GBP. It displays the data for asynchronous and changed requests on the Y axis and the number of times no subchannel was available on the Y2 axis. The report is generated from six RMF CF activity reports. The base reports are intervals at 10 o'clock on Tuesday in the time period from June 17 to July 29.

Figure 29 on page 124 shows the RMF CF Activity report for Tuesday, July 8. The cache structure MMDB2_GBP is shown at the bottom **F**. The spreadsheet graphic displays the REQUESTS - ASYNC, REQUESTS - CHNGD and DELAYED REQUESTS - NO SCH of the TOTAL row in the fourth bar in Figure 34 on page 134.

The chart shown in Figure 34 has the capability to display nearly all values from the structure activity section and usage summary section for a selected structure. You can also choose to display the data for one selected system or for all systems. This chart is part of report RepTrdStr in Figure 33 on page 133. This report is related to the reports RepIntStr and RepIntAct, which present data from the same sections of the RMF CF Activity report for a selected interval.

Asynchronous Requests and Delays

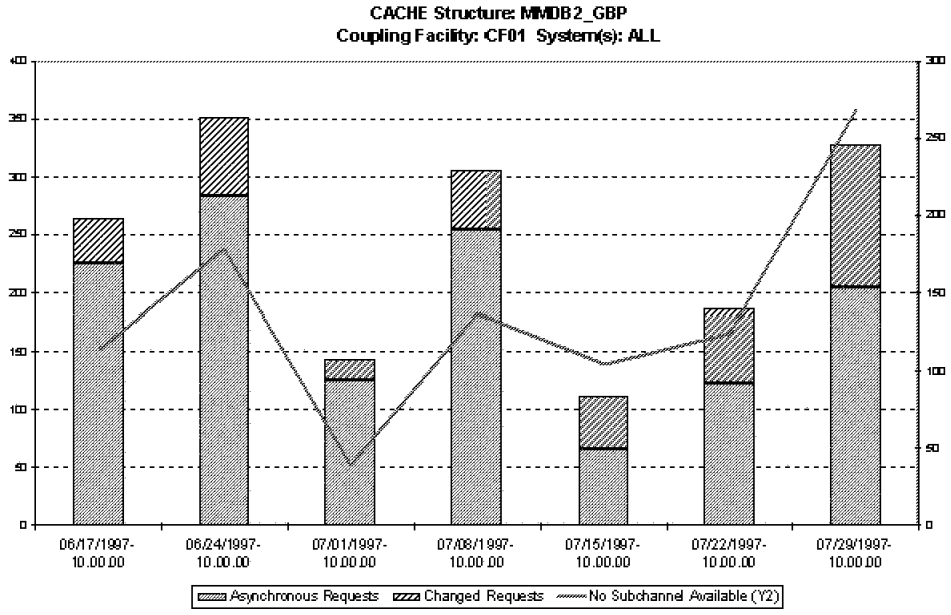


Figure 34. RMF CF Activity Spreadsheet - Structure MMDB2_GBP:

Report RepIntCF and the corresponding trend reports RepCFTrd, RepSubChn1, RepSubchn2, and RepCFSys present data for the CFs and the systems connected to the CFs. This data comes from the subchannel section of the RMF CF Activity report.

The reports not only present data from the RMF CF Activity report, they also calculate additional metrics from the data contained in the original RMF report.

Synchronous Request Utilization

One interesting value is engine utilization caused by synchronous requests processed by the OS/390 system. Synchronous request "spin" the sender CP as long as the CF processor executes them. Therefore, the utilization caused by these requests gives a basic idea of the data sharing overhead, not considering other effects. The number can be calculated directly from the subchannel section of the RMF CF activity report (see Figure 35 on page 135):

$$Percent_Engine_Util(System) = \frac{Sync_Reqs \times Sync_SrvTime}{IntvSecs \times 10000}$$

Where: (all references are to Figure 35 on page 135)

Sync_Reqs = REQUESTS - SYNC (**1**)
Sync_SrvTime = REQUESTS - SERVICE TIME(MIC) - SYNC (**2**)
IntvSecs = INTERVAL multiplied by 60 (**3**)
10000 scales everything to a percentage value

If you divide *Percent_Engine_Util(System)* by the number of CPs for the system, you get an idea how much CPU capacity is used to wait for the completion of synchronous requests.

```
                                C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                                                                                                PAGE  15
      OS/390                  SYSPLEX PRODPLEX           DATE 07/08/1997          INTERVAL 015.00.000 3
      REL. 02.04.00            RPT VERSION 2.4.0          TIME 10.00.00            CYCLE 01.000 SECONDS

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = CF01
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
                                                                SUBCHANNEL  ACTIVITY
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
          # REQ                                  ----------- REQUESTS -----------   -------------- DELAYED REQUESTS --------------
 SYSTEM   TOTAL                 --BUSY--             #   -SERVICE TIME(MIC)-          #     % OF  ------ AVG TIME(MIC) ------
 NAME     AVG/SEC -- CONFIG --  -COUNTS-             REQ     AVG    STD_DEV          REQ    REQ   /DEL    STD_DEV     /ALL

...
 K11      256184  SCH GEN    4  PTH 9035     SYNC  1  88681   171.5  2   433.4   SYNC   138   0.2%  914.7       2813      1.4
           284.6  SCH USE  6 4  SCH  138     ASYNC 4 164490  1069.9  5   1011    ASYNC 3280   2.0%  661.1       1495     13.2
                  SCH MAX    4               CHANGED     13  INCLUDED IN ASYNC  TOTAL 3418   1.4%
                  PTH        2               UNSUCC       0     0.0       0.0
 K12      440421  SCH GEN    4  PTH   10K    SYNC    240203   144.5     299.3   SYNC   727   0.3%  246.4      977.8      0.7
           489.4  SCH USE    4  SCH  727     ASYNC   194720   909.5      1255   ASYNC 5492   2.8%   1153       3034     32.5
                  SCH MAX    4               CHANGED     46  INCLUDED IN ASYNC  TOTAL 6219   1.4%
                  PTH        2               UNSUCC       0     0.0       0.0
```

Figure 35. RMF CF Activity Report - Subchannel Activity

Using the values annotated as **1**, **2** and **3** in Figure 35, we can calculate the utilization for system K11:

$$Percent_Engine_Util(K11) = \frac{88861 \times 171.5}{(15 \times 60) \times 10000} \approx 1.7\%$$

This value must be divided by the number of engines (or CPs) to get the CPU capacity for the system that is used to wait for the completion of the synchronous requests.

Report RepCFSys shows this value for all systems connected to the same CF and for all reporting intervals contained in the spreadsheet (see Figure 36 on page 136).

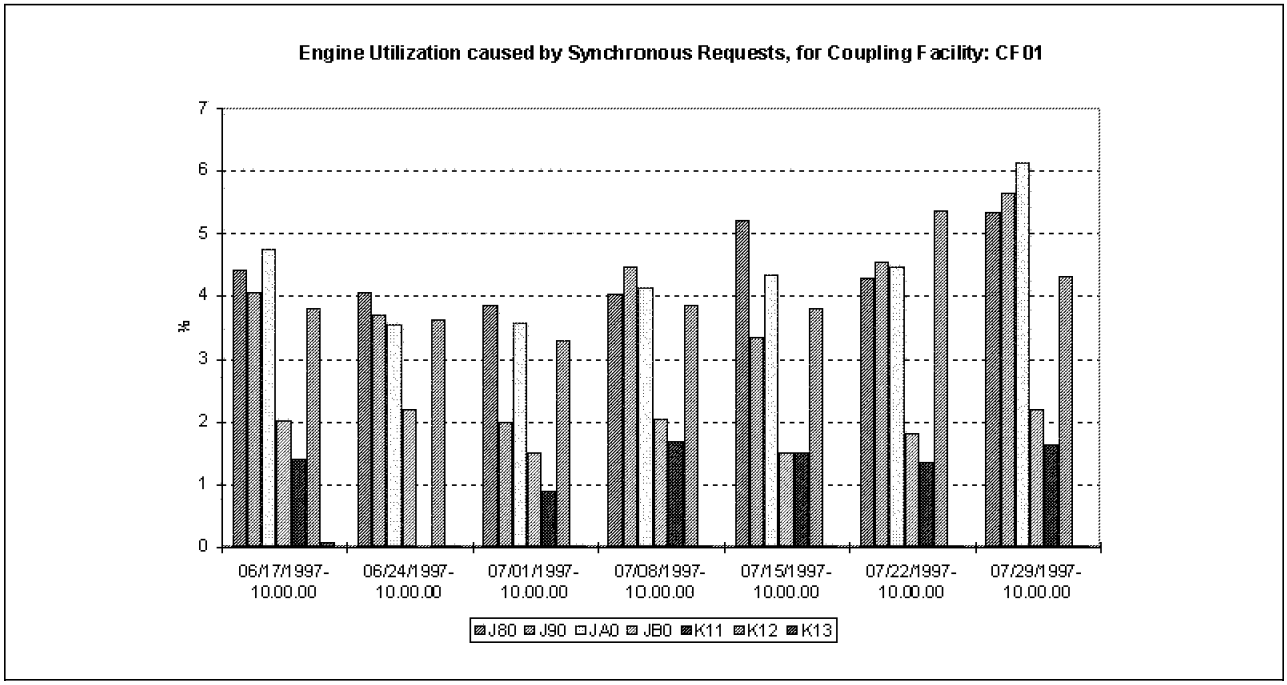


Figure 36. RMF CF Activity Spreadsheet - Engine Utilization for Synchronous Requests

Link Buffer Utilization

Another interesting value is the utilization of the links from the CEC to the CF. This value cannot be calculated directly from the RMF CF Activity report, but you can calculate the utilization of the link buffers. Section 4.3, "RMF Channel Activity Report" on page 137 shows how the link utilization can be obtained from the RMF Channel Activity report for new 9672 models.

Chapter 5, "A CF Processing Model" on page 143 discusses the relationship of software subchannels to link buffers. Using this relationship you can calculate the utilization of the link buffer by using the following equations:

$$Percent_Link_Buffer_Util(System) = \frac{Sync_Reqs \times Sync_SrvTime + Async_Reqs \times Async_SrvTime}{IntvSecs \times SubChn_InUse \times 10000}$$

Where: (all references are to Figure 35 on page 135)

Sync_Reqs = REQUESTS - SYNC (**1**)
Sync_SrvTime = REQUESTS - SERVICE TIME(MIC) - SYNC (**2**)
Async_Reqs = REQUESTS - ASYNC (**4**)
Async_SrvTime = REQUESTS - SERVICE TIME(MIC) - ASYNC (**5**)
IntvSecs = INTERVAL multiplied by 60 (**3**)
SubChn_InUse = CONFIG - SCH USE (**6**)
10000 scales everything to a percentage value

In a PR/SM environment with LPs sharing the same physical link, you have to summarize the numbers for all systems sharing the same link:

$$Percent_Link_Buffer_Util(CEC) = \sum_{i=1}^{n} Percent_Link_Buffer_Util(System_i)$$

Using the values in Figure 35 on page 135 we can now calculate the link buffer utilization for system K11:

$$Percent_Link_Buffer_Util(K11) = \frac{88681 \times 171.5 + 164490 \times 1011}{(60 \times 15) \times 4 \times 10000} \approx 5.3\%$$

Report RepCFSys again gives you the opportunity to display the value for all systems connected to the same CF and for all reporting intervals contained in the spreadsheet (see Figure 37). If multiple systems share the same links, you have to add the numbers shown in spreadsheet reports for these systems to get the link buffer utilization for the CEC.

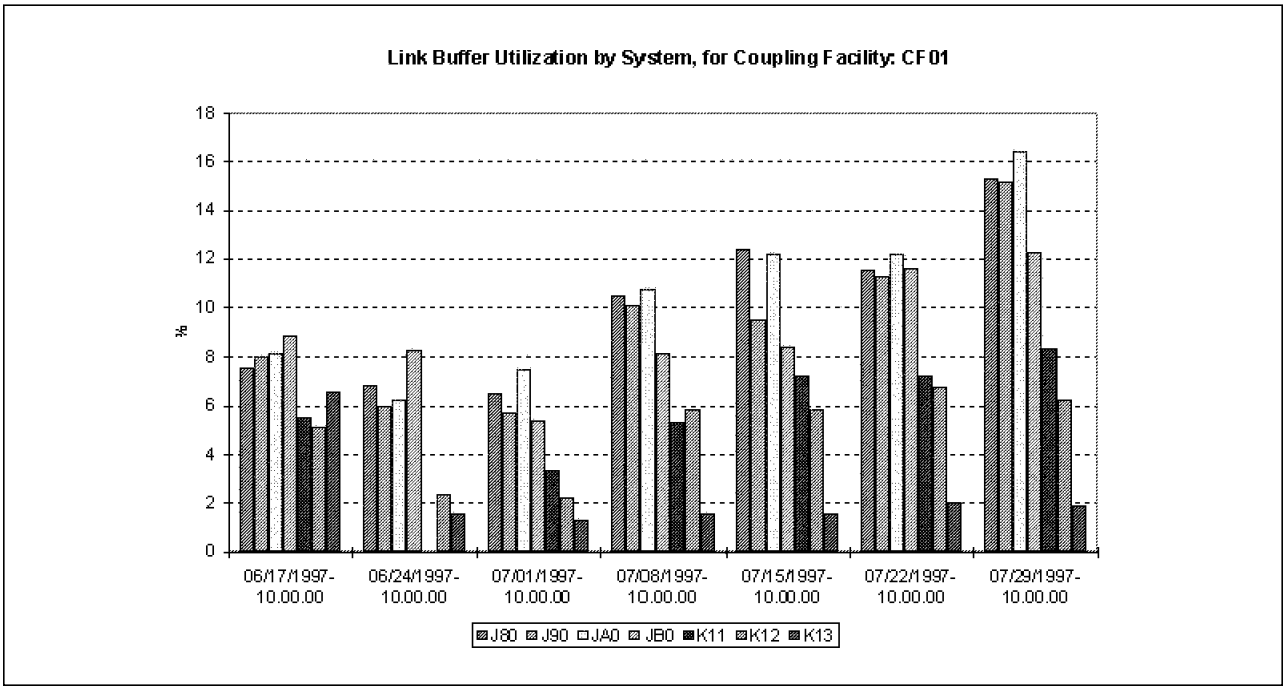


Figure 37. RMF CF Activity Spreadsheet - Link Buffer Utilization by System

4.3 RMF Channel Activity Report

Link Utilization

Since RMF 5.1, the RMF Channel Activity report shows intersystem channels. The values shown in the Channel Activity report were always zero in the past.

For 9672 G3 models with HiPerLinks and 9672 G4 models, the hardware supports the channel measurements and you can obtain the link utilization for each system directly from the RMF Channel Activity report. Figure 38 on page 138 shows an RMF Channel Activity report from a system supporting the new measurements. The channel type for CF links (annotated as **a** in Figure 38 on page 138) is printed as IS (InterSystem channel)[11]. In this example the utilization was about 5.38% for channel 09 (**b**) and 4.46% for channel A5 (**c**). The Channel Activity report shows the total utilization for all LPs sharing the same CEC.

For other models, R1 to G3 without HiPerLinks, the link utilization cannot be obtained directly.

```
                                C H A N N E L   P A T H   A C T I V I T Y
                                                                                PAGE    1
          OS/390                    SYSTEM ID SC66            DATE 08/20/1997          INTERVAL 04.59.999
          REL. 02.04.00             RPT VERSION 2.4.0         TIME 00.00.00            CYCLE 1.000 SECONDS

IODF = 05   CR-DATE: 08/10/97   CR-TIME: 15.59.09   ACT: ACTIVATE       MODE: LPAR        CPMF: AVAILABLE
CHANNEL PATH       UTILIZATION(%)          CHANNEL PATH       UTILIZATION(%)          CHANNEL PATH       UTILIZATION(%)
ID  TYPE  SHR    PARTITION    TOTAL        ID  TYPE  SHR    PARTITION    TOTAL        ID  TYPE  SHR    PARTITION    TOTAL
08   IS                        0.00        16   CN    Y     0.00          0.00        88   CN D  Y     0.26          3.05
09   IS  a                     5.38  b     17   CN    Y     0.00          0.00        89   CN    Y     0.00          0.00
10   CN D  Y     4.17         19.69        18   OS    Y     0.00          0.14        8A   CN    Y     0.00          0.00
11   CC D  Y     0.00          3.68        80   OS    Y     0.00          0.00        8B   CN    Y     0.00          0.00
12   CN    Y     0.00          0.00        84   CN D  Y     2.01         18.20        8C   CN D  Y     0.28          2.62
13   CN    Y     0.00          0.00        85   CN D  Y     0.00          0.00        8D   CN    Y     0.00          0.00
14   CN D  Y     3.19          3.61        86   CN    Y     0.00          0.00        8E   CN    Y     0.00          0.00
15   CN    Y     0.00          0.00        87   CN    Y     0.00          0.00        8F   CN    Y     0.00          0.00
90   BL        OFFLINE                   9A   CN    Y     0.00          0.00        A8   CN D  Y     4.21         22.03
91   BL        OFFLINE                   9B   CN    Y     0.00          0.00        A9   CN D  Y     0.00          0.00
92   BL        OFFLINE                   9C   OS    Y     0.00          0.00        AA   CN    Y     0.00          0.00
94   BL        OFFLINE                   A0   BL        OFFLINE                     AB   CN    Y     0.00          0.00
95   BL        OFFLINE                   A1   BL        OFFLINE                     AC   CN D  Y     3.22          3.68
96   BL        OFFLINE                   A2   BL        OFFLINE                     AD   CN    Y     0.00          0.00
98   CN D  Y     0.00          4.67        A4   IS                        0.00        AE   CN    Y     0.00          0.00
99   CN    Y     0.00          0.00        A5   IS                        4.46  c     AF   CN    Y     0.00          0.00
B0   CN D  Y     0.32          2.48        BB   CN    Y     0.00          0.00        DF   CN    Y     0.00          0.00
B1   CN    Y     0.00          0.00        BC   BL        OFFLINE                     E0   OS    Y     0.00          0.00
B2   CN    Y     0.00          0.00        BD   BL        OFFLINE                     E4   CN D  Y     0.34          3.19
B3   CN    Y     0.00          0.00        BE   BL        OFFLINE                     E5   CN    Y     0.00          0.00
B4   OS    Y     0.00          0.00        D8   OS    Y     0.00          0.00        E6   CN    Y     0.00          0.00
B8   CC D  Y     0.00          3.75        DC   CN D  Y     2.19         20.68        E7   CN    Y     0.00          0.00
B9   CN    Y     0.00          0.00        DD   CN D  Y     0.00          4.53        E8   BL        OFFLINE
BA   CN    Y     0.00          0.00        DE   CN    Y     0.00          0.00        E9   BL        OFFLINE
EA   BL        OFFLINE
EC   BL        OFFLINE
ED   BL        OFFLINE
EE   BL        OFFLINE
```

Figure 38. RMF Channel Activity Report

You can also view the link utilization online in the RMF Monitor II Channel Activity report. The information shown in the online report is the same as in the Postprocessor report, only for small time intervals.

4.4 RMF XCF Activity Reports

The XCF Activity report is produced using the RMF control statement REPORTS(XCF) in the Postprocessor. The following examples are an extract from the XCF Activity report, which is divided into the three sections shown.

The *Usage by System* section gives information about messages sent to and received from each remote system in the sysplex, broken down by transport class. Use this section to check the class lengths and message buffer space parameters.

q shows what percentage of the messages sent fit into the defined buffer. Ideally, you would want a 100% fit. Defining too big a buffer is a waste of space. Defining a buffer that is too small incurs overhead as XCF must find a buffer big enough to fit the message. The %OVR field indicates the percentage of *big* buffers that suffered performance degradation. Notice that for system SC47, not all the big buffers caused performance degradation.

Be aware of non-zero values in ALL PATHS UNAVAIL **r** and REQ REJECT **s** as this will also cause performance degradation.

[11] Notice that the report is from a different system than the reports in the examples discussed in the previous chapters.

```
                                    X C F   A C T I V I T Y                                                    PAGE    1

          OS/390                   SYSTEM ID SC62           DATE 03/12/1997          INTERVAL 10.00.000
          REL. 01.03.00            RPT VERSION 1.3.0        TIME 16.00.00            CYCLE 1.000 SECONDS

                                    XCF USAGE BY SYSTEM

-------------------------------------------------------------------------------------------------------------------------------
                                    REMOTE SYSTEMS                                                              LOCAL
-------------------------------------------------------------------------------------------------------------------------------
                        OUTBOUND FROM SC62                                    INBOUND TO SC62                   SC62
-------------------------------------------------------------------------------------------------------------------------------
                                        ----- BUFFER -----   r ALL    s                              s
TO        TRANSPORT  BUFFER      REQ   % q %    %    %     PATHS     REQ   FROM          REQ      REQ   TRANSPORT     REQ
SYSTEM    CLASS      LENGTH      OUT   SML  FIT  BIG  OVR  UNAVAIL  REJECT SYSTEM        IN   REJECT   CLASS      REJECT
SC42      DEFAULT       956    1,772    0   97    3  100        0       0  SC42       1,966        0   DEFAULT         0
SC43      DEFAULT       956    1,923    0   97    3  100        0       0  SC43       1,397        0
SC47      DEFAULT       956    2,540    0   91    9   68        0       0  SC47       1,379        0
 ...
SC55      DEFAULT       956    1,960    0   97    3  100        0       0  SC55       1,265        0
SC61      DEFAULT       956    1,662    0   97    3  100        0       0  SC61       1,353        0
                             ----------                                              ----------
TOTAL                          26,565                                    TOTAL       15,847
```

Figure 39. RMF XCF Activity Report - Usage by System

The *Usage by Member* section gives information about messages sent to and from each remote system, broken down by remote group and member, and summarizes messages sent and received by the local system (the local system is the system on which the data was collected), broken down by local group and member. Use this section to check message traffic loads associated with groups **t** and members **v**, and check for groups that are candidates to be put in their own transport classes.

```
                                    X C F   A C T I V I T Y                                                    PAGE    2

          OS/390                   SYSTEM ID SC62           DATE 03/12/1997          INTERVAL 10.00.000
          REL. 01.03.00            RPT VERSION 1.3.0        TIME 16.00.00            CYCLE 1.000 SECONDS

                                    XCF USAGE BY MEMBER

-------------------------------------------------------------------------------------------------------------------------------
              MEMBERS COMMUNICATING WITH SC62                                          MEMBERS ON SC62
-------------------------------------------------------------------------  ----------------------------------------------
                                                 REQ          REQ
  t               v                              FROM         TO                                                REQ          REQ
ISTXCF    SC42M$$$USIBMSC     SC42               93           94          ISTXCF    SC62M$$$USIBMSC         1,038        1,037
          SC43M$$$USIBMSC     SC43               95           94                                          ----------   ----------
          SC47M$$$USIBMSC     SC47               98           99          TOTAL                           1,038        1,037
 ...
          SC55M$$$USIBMSC     SC55               93           94
          SC61M$$$USIBMSC     SC61               93           92
                                                ----------   ----------
TOTAL                                            1,038        1,037
 .
 .
SYSGRS    SC42                SC42              282          227          SYSGRS    SC62                      637          582
          SC43                SC43                0            0                                          ----------   ----------
          SC47                SC47                2            2          TOTAL                             637          582
 ...
          SC55                SC55                0            0
          SC61                SC61                0            0
                                                ----------   ----------
TOTAL                                              637          582
```

Figure 40. RMF XCF Activity Report - Usage by Member

The *Path Statistics* section describes messages sent to and from each remote system, broken down by signalling path. Use this report to determine whether the number of XCF signalling paths is sufficient for the message traffic.

Both Coupling Facility structure and CTC signalling paths are reported as shown in **w**. A value of *UNKNOWN is an indication of an incomplete, incorrect or

inactive CTC definition. **x** and **y** give an indication of how often an XCF signal had to be retried.

- BUSY: The number of times XCF selected a signalling path while a message was already in the process of being transferred.
- RETRY: The number of times XCF initialized the signalling path.
- BUFFERS UNAVAIL: The number of times that XCF was not able to get an inbound message buffer for the signalling path in anticipation of receiving a new message.

Note: If the XCF system, path, or member becomes inactive during the RMF interval, the appropriate counters are reinitialized. This is indicated in the report by the message COUNTS RESET.

```
                                        X C F  A C T I V I T Y                                                  PAGE   15
           OS/390                  SYSTEM ID SC62            DATE 03/12/1997          INTERVAL 10.00.000
           REL. 01.03.00           RPT VERSION 1.3.0         TIME 16.00.00            CYCLE 1.000 SECONDS

TOTAL SAMPLES = 596                                  XCF PATH STATISTICS
-------------------------------------------------------------------------------------------------------------------------------
                            OUTBOUND FROM SC62                                                INBOUND TO SC62
-------------------------------------------------------------------------------------------------------------------------------
         T FROM/TO                                                                    T FROM/TO                          y
 TO      Y DEVICE, OR      TRANSPORT        REQ     AVG Q                  x        FROM     Y DEVICE, OR          REQ BUFFERS
 SYSTEM  P STRUCTURE       CLASS            OUT     LNGTH    AVAIL    BUSY   RETRY  SYSTEM   P STRUCTURE            IN UNAVAIL
 SC47    S IXC_DEFAULT_1   DEFAULT           25      0.01       25       0       0  SC47     S IXC_DEFAULT_1       716       0
         S IXC_DEFAULT_2   DEFAULT           61      0.01       60       1       0           S IXC_DEFAULT_2       737       0
         C 5050 TO 4330    DEFAULT        1,578      0.01    1,578       0       0           C 5330 TO 4050      1,817       0
         C 5058 TO 4338    DEFAULT        1,625      0.02    1,625       0       0           C 5338 TO 4058      2,431       0
 ....
 SC55    S IXC_DEFAULT_1   DEFAULT           23      0.01       21       2       0  SC55     S IXC_DEFAULT_1       838       0
         S IXC_DEFAULT_2   DEFAULT           36      0.01       36       0       0           S IXC_DEFAULT_2       702       0
         C 5040 TO 4330    DEFAULT        1,198      0.01    1,197       1       0           C 5330 TO 4040      1,436       0
                                        X C F  A C T I V I T Y                                                  PAGE   16
           OS/390                  SYSTEM ID SC62            DATE 03/12/1997          INTERVAL 10.00.000
           REL. 01.03.00           RPT VERSION 1.3.0         TIME 16.00.00            CYCLE 1.000 SECONDS

TOTAL SAMPLES = 596                                  XCF PATH STATISTICS
-------------------------------------------------------------------------------------------------------------------------------
                            OUTBOUND FROM SC62                                                INBOUND TO SC62
-------------------------------------------------------------------------------------------------------------------------------
         T FROM/TO                                                                    T FROM/TO
 TO      Y DEVICE, OR      TRANSPORT        REQ     AVG Q                            FROM     Y DEVICE, OR          REQ BUFFERS
 SYSTEM  P STRUCTURE       CLASS            OUT     LNGTH    AVAIL    BUSY   RETRY  SYSTEM   P STRUCTURE            IN UNAVAIL
 SC55    C 5048 TO 4338    DEFAULT        1,448      0.02    1,448       0       0  SC55     C 5338 TO 4048      1,822       0
 (CONT)                                                                               (CONT)
 SC61    S IXC_DEFAULT_1   DEFAULT           38      0.01       38       0       0  SC61     S IXC_DEFAULT_1       710       0
         S IXC_DEFAULT_2   DEFAULT           58      0.01       58       0       0           S IXC_DEFAULT_2       688       0
  w      C 5320 TO 4330    DEFAULT          744      0.02      744       0       0           C 5330 TO 4320      1,626       0
         C 5328 TO 4338    DEFAULT        1,573      0.01    1,573       0       0           C 5338 TO 4328      1,572       0
*UNKNOWN C 5010 TO         DEFAULT            0      0.00        0       0       0 *UNKNOWN  C      TO 4010            0       0
         C 5018 TO         DEFAULT            0      0.00        0       0       0           C      TO 4018            0       0
         C 5020 TO         DEFAULT            0      0.00        0       0       0           C      TO 4020            0       0
         C 5028 TO         DEFAULT            0      0.00        0       0       0           C      TO 4028            0       0
         C 5130 TO         DEFAULT            0      0.00        0       0       0           C      TO 4130            0       0
         C 5138 TO         DEFAULT            0      0.00        0       0       0           C      TO 4138            0       0
         C 5330 TO         DEFAULT            0      0.00        0       0       0           C      TO 4330            0       0
         C 5338 TO         DEFAULT            0      0.00        0       0       0           C      TO 4338            0       0
                                           ----------                                                              ----------
 TOTAL                                       34,811                                TOTAL                          60,249
```

Figure 41. RMF XCF Activity Report - Path Statistics

4.5 RMF OMVS Kernel Activity Report

RMF reports give you information about the OMVS kernel activity and provide the following information: Figure 42 shows a sample output.

```
                                    O M V S   K E R N E L   A C T I V I T Y
                                                                                                                    PAGE    1
          OS/390                  SYSTEM ID SC52            DATE 07/29/1997          INTERVAL 10.00.000
          REL. 01.03.00           RPT VERSION 1.3.0         TIME 09.40.00            CYCLE 1.000 SECONDS
TOTAL SAMPLES =   598
                                                  OMVS SYSTEM CALL ACTIVITY 1
-------------------------------------------------------------------------------------------------------------------------------
                 MINIMUM  AVERAGE  MAXIMUM
-------------------------------------------------------------------------------------------------------------------------------
 SYSCALLS (N/S)   5.000    41.03    532.0
 CPU TIME (H/S)   0.000    0.048    1.000
                                                  OMVS PROCESS ACTIVITY 2
-------------------------------------------------------------------------------------------------------------------------------
                          PROCESSES                        USERS                  PROCESSES PER USERS
 MAXIMUM  (TOT)             300                              50                      10125
-------------------------------------------------------------------------------------------------------------------------------
                 MINIMUM  AVERAGE  MAXIMUM      MINIMUM  AVERAGE  MAXIMUM      MINIMUM  AVERAGE  MAXIMUM
-------------------------------------------------------------------------------------------------------------------------------
 CURRENT  (TOT)      33    33.00       33            0    0.000        0
 OVERRUNS (N/S)   0.000    0.000    0.000        0.000    0.000    0.000        0.000    0.000    0.000
                                               OMVS INTER-PROCESS COMMUNICATION 3
-------------------------------------------------------------------------------------------------------------------------------
                    MESSAGE QUEUE IDS              SEMAPHORE IDS              SHARED MEMORY IDS            SHARED MEMORY PAGES
 MAXIMUM  (TOT)             20000                          20000                      20000                        2621K
-------------------------------------------------------------------------------------------------------------------------------
                 MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM
-------------------------------------------------------------------------------------------------------------------------------
 CURRENT  (TOT)    5.000    4.983    5.000      46.00    45.85    46.00      2.000    1.993    2.000       8013    1.993     8013
 OVERRUNS (N/S)    0.000    0.000    0.000      0.000    0.000    0.000      0.000    0.000    0.000      0.000    0.000    0.000
                                                     OMVS MEMORY MAP 4
-------------------------------------------------------------------------------------------------------------------------------
                 MEMORY MAP STORAGE PAGES        SHARED STORAGE PAGES
 MAXIMUM  (TOT)             4096                           32.8M
-------------------------------------------------------------------------------------------------------------------------------
                 MINIMUM  AVERAGE  MAXIMUM    MINIMUM  AVERAGE  MAXIMUM
-------------------------------------------------------------------------------------------------------------------------------
 CURRENT  (TOT)    429.0    427.6    429.0      97083    96752    97083
 OVERRUNS (N/S)    0.000    0.000    0.000      0.000    0.000    0.000
 Units:   (TOT) = Total Value, (N/S) = Number per Second, (H/S) = Hundredths of seconds per Second
```

Figure 42. Sample RMF Report Output for OMVS Kernel Activity

1 OMVS System Call Activity

This section of the report gives the rate of syscalls executed in the OMVS address space. Syscalls are executed on behalf of OS/390 UNIX Services applications (3.5.4, “Services Executed Using PC to OMVS Address Space” on page 112). Note that the CPU TIME field gives the time spent in hundredths of seconds per second.

2 OMVS Process Activity

The first part of this section repeats the current settings of MAXPROCSYS, MAXUIDS, MAXPROCUSER. The second part shows the behavior of these values in this current interval. If these values have been changed during the interval, *** is displayed instead.

3 OMVS Inter-Process Communication

Again the current settings are repeated. In this case we have the values of IPCMSGNIDS, IPCSEMNIDS, IPCSHMNIDS, and IPCSHMSPAGES.

4 OMVS Memory Map

The two values given here reflect the current values of MAXMMAPAREA and MAXSHAREPAGES.

Chapter 5. A CF Processing Model

At a Glance

This chapter introduces an analytic model for use in analyzing usage of an existing CF and modeling the effect of future changes in activity or configuration (or both).

The input for the model can be obtained from RMF CF Activity Reports.

Recommended Reading:

- Arnold O. Allen: *Probability, Statistics, and Queueing Theory with Computer Science Applications*

© Copyright IBM Corp. 1998

5.1 Introduction to Using the CF Processing Model

This chapter introduces an analytic model for use in analyzing usage of an existing CF and modeling the effect of future changes in activity or configuration (or both).

The model we describe depends on the sending CEC, the link type and the receiving CF. Each of these components makes a significant contribution to overall processing of requests to the CF.

This model has been validated only for IBM CECs and CFs. If the model is to be used for any non-IBM CEC, it is particularly important to establish how much CPU time is spend on the sender CEC because there is no direct way of calculating this figure from RMF. Similarly, although it may be possible to obtain non-IBM CF CP times from RMF, the tool has no way of projecting behavior for any different non-IBM CF.

Although in any particular run of the tool a single sending CEC and a single CF are modeled, multiple runs can be used to model the CF usage in a multi-CEC, multi-CF sysplex. At this stage, the tool does not model shared CF CPs.

Using the tool involves:

1. Obtaining input from RMF
2. Calibrating the model
3. Running the tool for each scenario to be modeled

The tool can be used to model the following changes:

1. Increasing the CF activity rate
2. Increasing the CF utilization
3. Changing the number of CF links
4. Changing the CF link speed
5. Adding a CF CP (or migrating to another CF)
6. Changing the sender CEC

The tool can also model combinations of these changes.

5.2 Description of the CF Processing Model

This section describes some important considerations for understanding CF capacity.

We discuss the following topics:

- CF request processing
- CF times, delays, and queues
- A simple mathematical model for CF request processing

5.2.1 CF Request Processing

In principle, a CF request is initiated by OS/390(XES), transferred to the CF if there is a free CF link, and processed by a CF CP. When the CF request has been processed, data and other information is sent back to OS/390 via the CF link.

Figure 43 on page 145 shows one CEC with two logical partitions running OS/390. The CEC is connected to the CF via two CF links. The CF has one CP.

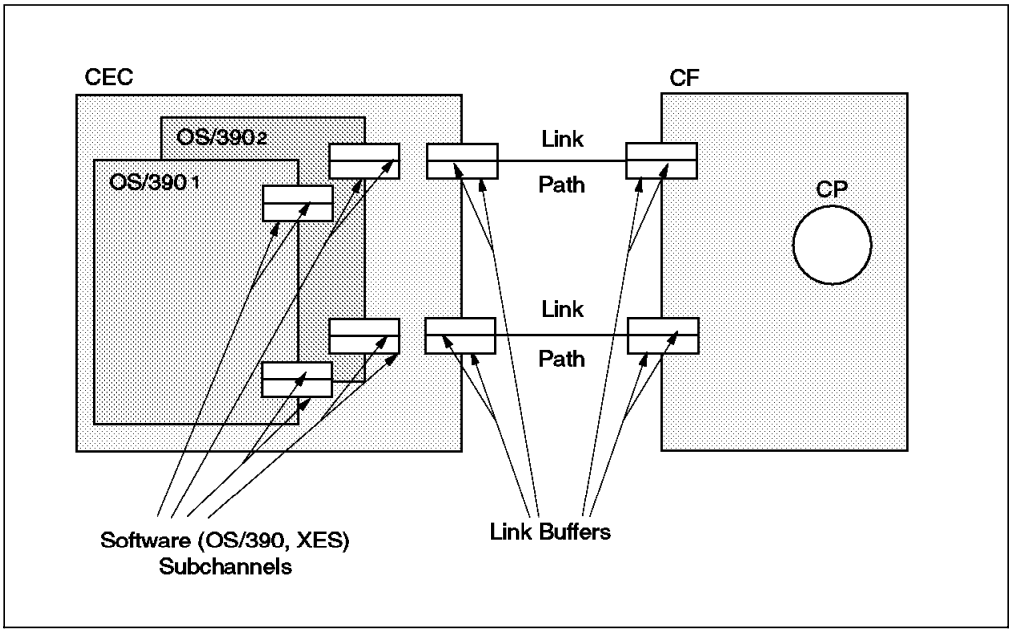


Figure 43. Some of the Components Participating in a CF Request

Explanations related to the components in Figure 43 follow:

Link/Path: Links connect the CEC and the CF. There are two links in this picture. The term path is used by RMF and is equivalent to a link.

Link Buffer: The link buffer acts as a buffer for the CF requests. There are two buffers per link. RMF does not report the activity related to these buffers.

Software Subchannel: The software subchannel acts as a buffer for the CF requests on OS/390. There are two buffers per link in each OS/390. There is a one-to-one mapping to the link buffer.

Buffers in CF: The buffers in the CF are used to hold the CF request on the CF side. There are two buffers per link, as on the CEC side.

CF Request Delays

Before describing the actions involved in processing the CF request, we list different delay reasons for the CF requests. OS/390 places the request in a subchannel to initiate its transfer to the CF. The subchannel may be unable to accept the request for the following reasons:

Software Subchannel Busy

- All software subchannels in this OS/390 are busy waiting for one or more of the following:
 - An available link buffer (EMIF contention)
 - Data transfer in the link to complete
 - A CF processing the request

Link Buffer Busy

- At least one software subchannel is free, but the corresponding link buffer is busy (EMIF contention), because it is waiting for the following:
 - Data transfer in the link to complete
 - A CF to complete the request

The different types of CF requests are processed and delays handled as follows:

1. If the CF request cannot be started in OS/390 because all its subchannels or link buffers are busy, then the following actions are taken as shown in Figure 44 on page 147, depending on the request:

Synchronous Immediate Request

- A synchronous immediate request (for example, a lock-type request) "spins" on the CEC CP until a subchannel becomes available if:
 - The software subchannel is busy **1**
 - The link buffer is busy **2**

Synchronous Non-Immediate Request

- A synchronous request (non-immediate) is:
 - Changed to asynchronous and queued in OS/390 if the software subchannel is busy **3**
 - Restarted as an asynchronous request and queued in OS/390 if the link buffer is busy **4**

Asynchronous Request

- An asynchronous request is queued in OS/390 if:
 - The software subchannel is busy **5**
 - The link buffer is busy; the request is started again as a new request (redriven) **6**

The queued request in OS/390 is redriven when a subchannel or buffer becomes available.

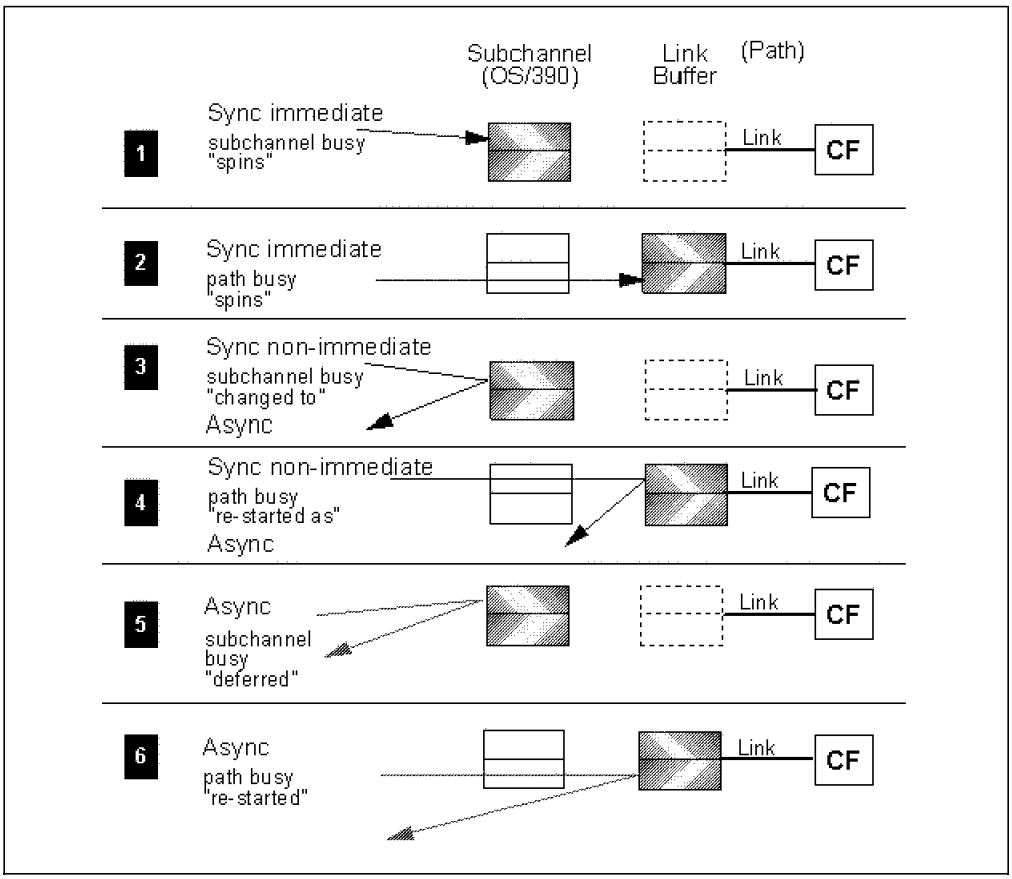


Figure 44. CF Requests and Delay Types

2. When one of the subchannels or buffers is free, the request is started (by XES or by PR/SM) and the commands and/or data are transferred to the CF, as soon as the link is free. There are two subchannels (and link buffers) per link, so the link may be busy transferring data on the other link buffer. Data transfer does not need a free CP in the CF. The request is queued in the CF if the CF does not have an available CP. This subchannel (and buffer) stays reserved for this request until it is completed. However, the link is released when the request arrives to the CF.
3. The CF processes the request as soon as it has a free CP.
4. When the CF has processed the request, the response is sent back to OS/390 as soon as the link is free. There are two subchannels (and link buffers) per link, so the link may be busy transferring data on the other buffer. As the subchannel (and buffer) has been reserved for this request, the request uses the same link as before.

5.2.2 CF Times, Delays and Queues

For the actions of the CF requests in 5.2.1, “CF Request Processing” on page 144 for points 1 to 4, the delays and processing are as follows:

1. Wait for the CF or the link.

 If the subchannels (buffers) of the CF links are busy, either the request is put on a queue and redriven, or it “spins” on the CEC CP, depending on the type of request.

2. Send a request through the link.

 The transfer time for a lock request (approximately 250 bytes) is only a few microseconds (µs). For a 4KB page on a 50 MB/s link, the transfer time is about 80 µs, and about 40 µs for on a 100 MB/s link. We assume that this is the total link processing time.

3. The CF processes the request.

 The service time is a function of the CF CP speed and the type of request. Its average is observed to be less than 100 µs (if mostly lock-type requests) or more than 200 µs (if mostly DB2 cache-type requests) on an R5-based CF. The service time distribution may be assumed to be close to constant if lock-type requests are predominant, exponential if DB2 cache-type requests are predominant, and hyperexponential if both of these are mixed.

 There may be queueing in the front of a CP when the requests come from several links, from several subchannels (buffers), and from several requestors.

4. Send the response back through the link.

 There is no queueing or waiting for a subchannel (buffer), but there may be some wait time for the link.

 The transfer time for a lock request (approximately 250 bytes) is only a few microseconds. For a 4KB page on a 50 MB/s link, the transfer time is about 80 µs, and about 40 µs on a 100 MB/s link. We assume that this is the total link processing time.

5.2.3 A Model for the CF Request

Because we are interested in the capacity of the CF, we approach its performance characteristics by creating a simplified model of them. Using this model, we try to find out how much the components of the CF are loaded.

Figure 45 shows the elements of the model.

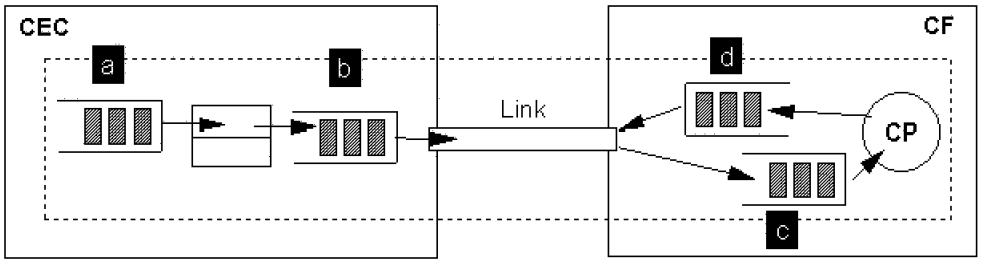


Figure 45. A Model for the CF Request

The processing phases of the model are as follows:

1. The request either arrives at the queue **a** to the subchannels (buffers) or "spins" on the subchannel (buffer).
2. The request obtains a subchannel (buffer) and it is served in the CEC.
3. The request is queued **b** to the link.
4. The request is served (transferred on the link).
5. The request arrives at the queue **c** waiting for the CF CP.
6. The request is served at the CF CP.
7. The request arrives at the queue **d** to the link.
8. Finally, the request is served (receives transfer) on the link.

It is best to do the calculations in phases. We demonstrate how you can base this on values from RMF reports. We apply a queueing model three times.

1. We calculate the processing time of the request in the CF CP. This includes the queue time to the CF CP **c** (which we estimate with a queuing model), and the service time in the CF CP (which is obtained from RMF).
2. We obtain the link buffer busy percentage and calculate the request arrival rate from RMF reports, and then estimate the link service time and queue time with a queueing model, **b** and **d**.
3. We sum up the previous time components and now have an estimate of the total service time. This time is compared to the reported service time from RMF reports.
4. We estimate the queue time **a** to the link buffers by using the queueing model again.

We can now estimate the total time as the total service time plus the queueing time calculated in step 3. This should compare reasonably well with the total time found from RMF; in other words, the weighted average CF service time. Now the model can be used to estimate different conditions because it estimates queueing where it can occur.

5.2.3.1 Queueing Formulas for CF CP Processing Time Calculations

For the CF time calculations, we use a generalization of the M/M/C queueing model (exponential arrival distribution, exponential service time distribution, one or more servers), as described by Allen and Cunneen. This formula extends the M/M/C model to G/G/C models where the arrival distribution or the service time distribution, or both, are clustered (hyperexponential). The G/G/C queueing model refers to general inter-arrival time distribution, general service time distribution, and one or more servers.

Refer to Figure 43 on page 145 and Figure 45 on page 148 for the elements and the queueing model of the CF. The formula for calculating the queue time is as follows:

$$T_c = \frac{C_a^2 + C_s^2}{2} \times \frac{\dfrac{(U \times C)^C}{C!}}{\dfrac{(U \times C)^C}{C!} + (1 - U) \times \displaystyle\sum_{n=0}^{C-1} \frac{(U \times C)^n}{n!}} \times \frac{}{C \times (1 - U)} \times S$$

where:

T_c = Queue time
U = CF CP utilization
C = Number of CPs in the CF
S = Service time for the CF CP

$$\frac{C_a^2 + C_s^2}{2} = k =$$ Allen-Cunneen approximation factor for G/G/C

C_a^2 = Coefficient of the arrival distribution

C_s^2 = Coefficient of the service time distribution

$$C^2 = \frac{(\text{Standard deviation})^2}{(\text{Average})^2}$$

k=0.5 for exponential arrival and constant service
k=1.0 for exponential arrival and exponential service
k=4.0 when both arrival and service are hyperexponential, with standard deviation being two times the average

Because the general formula is so complex, we use another approximation that is accurate for cases when the CF has less than three CPs. The estimates of queue times in the approximation are reasonably good in other cases. When the queue time is over half the service time and there are four or five CF CPs, the approximation is low but within 10% of the general formula. When there are between six and ten CF CPs, the approximation is low but within 20% of the general formula. In general, the longer the queue time, the closer the approximation.

In our approximation (accurate for one and two CPs in the CF), the queue time (T_c) for the request is as follows:

$$T_c = k \times \frac{U^C}{1 - U^C} \times S$$

If we assume that lock requests arrival time distribution is exponential and that the service time distribution is between constant to exponential, then for a CF doing mostly locking, k<1.

For DB2 GBP requests, if we assume that both the arrival and the service time distributions are exponential to hyperexponential, then for a CF processing mostly requests of this type, k>1.

RMF Input: RMF CF Summary report provides us the values for the CF request rate R_{cf} (req/s), CF CP utilization U (%) and the number of CF CPs (C). The CF service time S can be calculated as:

$$S = \frac{C \times U}{R_{cf}}$$

In the Usage Summary report, shown in Figure 28 on page 123 at point **9**, we can find the CP utilization. Additionally, the same RMF report gives the number of requests to the structures in the column AVG REQ/SEC, which is just before the column at point **6**.

For example, in Figure 28 on page 123 we have U=40.7%, one CP and R_{cf}=3370.2 req/sec. Then:

$$S = \frac{1 \times 0.407}{3370.2 \text{ req/sec}} = 120\,\mu s$$

Example: Assume that we have a one-CP CF which is 40% utilized and the service time is 100 µs. Assuming exponential service time distributions (k=1), the queue time (T_c) is:

$$T_c = 1.0 \times \frac{0.4}{1 - 0.4} \times 100\,\mu s = 67\,\mu s$$

The processing time in the CF, including the queue time, is:

CF time $= 67\,\mu s + 100\,\mu s = 167\,\mu s$

5.2.3.2 Link Service and Queue Time Calculations

If we have HiPerLinks, we use the RMF Channel Activity report (not the CF Activity reports) to find the link busy percentage (U_l). If we do not have HiPerLinks, we have to estimate the link service time and the link busy percentage. The links are marked as type IS in the report. Refer to points **a** and **b** in Figure 38 on page 138. You have to select the links that connect to this CF.

RMF Input: In our case, assume we find values whose average is:

$U_l = 10\%$

We calculate the link service time based on the link busy percentage, as follows:

$$S_l = \frac{U_l}{R_a \,/\, \text{Number of links}}$$

where R_a is the arrival rate of the requests.

To calculate the queue time to the link, we use the queueing model Allen-Cunneen generalization of M/M/1 (exponential arrival, exponential service, one server), as we did for CF CP calculation. The service time on the links will vary from less than one page transfer time (lock request) to 32 (or even more) page transfer time. This is a single server queue. This link buffer is assigned to one link. That one link is the server. The formula is as follows:

$$T_b = k \times \frac{U_l}{1 - U} \times S_l$$

where:

- T_b = Queue time to the link
- U_l = Link utilization
- S_l = Link service time
- k = 1.0 for exponential distributions
- k > 1.0 for hyperexponential distributions

Example: Suppose we have a CEC issuing 2222 CF requests/second. Assume there are two links from the CEC to the CF. Then in our example the link service time will be:

$$S_l = \frac{0.10}{2222 \,/\, 2} = 88\,\mu s$$

And the link queue time will be

$$T_b = 1.0 \times \frac{0.1}{1 - 0.1} \times 88\,\mu s = 10\,\mu s$$

This queue time is the same in both directions on the link.

$T_d = T_b = 10\,\mu s$

5.2.3.3 Total CF Request Service Time Calculation

We are now almost ready to calculate the total CF service time. Handling a CF request requires CEC service time from the sender, and this CEC service time (S_{cec}) is heavily dependent on the architecture and speed of the CEC. It is also dependent on the type of requests (synchronous or asynchronous) and requestors (IMS, DB2, RACF and so on). The tool KGTV calculates (S_{cec}) for different CECs.

Example: We assume in our example that the CEC service time (S_{cec}) is:

$S_{cec} = 80\,\mu s$

To get the total service time, S_{tot}, for the requests, we sum up the previous time components:

$S_{tot} = T_b\,(\text{send}) + T_c\,(\text{CF}) + S_{cf}\,(\text{CF}) + T_d\,(\text{receive}) + S_l\,(\text{Link}) + S_{cec}$

Example: In our example, the total service time is:

$S_{tot} = 10\,\mu s + 67\,\mu s + 100\,\mu s + 10\,\mu s + 88\,\mu s + 80\,\mu s = 355\,\mu s$

This time should be about the same weighted average service time (S_{tot}) as calculated from the RMF reports at “Link Buffer Busy Percentage Calculation” on page 153. We have the value of 400 µs in “Example” on page 153. This means that our model gives overly optimistic results: our number is about 10% smaller than the number reported by RMF.

Model Calibration: To adjust our model, we can modify the various k factors. Making them larger means that the distributions are more clustered. A value of k=4, for example, means that we assume that the standard deviation is two times the average in both arrival and service time distributions. Then we recalculate the values.

We can make projections based on the figures we have just calculated, but you should remember that the model is optimistic; that is, it underestimates by about 10%.

5.2.3.4 Link Buffer Queue Time and Total Time Calculations

The link buffer queueing time has to be calculated to get the total OS/390 XES time for the CF requests.

We calculate the queue time using the Allen-Cunneen extension to the queueing model M/M/C for the third time. The formula is as follows:

$$T_a = k \times \frac{U_b^C}{1 - U_b^C} \times S_{tot}$$

where:

T_a = Queue time to the link buffer
U_b = Link buffer utilization

$S_{tot.}$ = Total CF service time
C = Number of link buffers
k = 1.0 for exponential distributions
k > 1.0 for hyperexponential distributions

Link Buffer Busy Percentage Calculation: We use the RMF Subchannel Activity report to calculate the link buffer busy percentage (U_b). We have to select the group of links that is used by the OS/390 system we are looking at. In case of other systems sharing the same links (EMIF), we have to calculate the requests from all systems as one entity. Based on that entity in the report, we have to calculate the CF request rate (R_a) and the total weighted average service time (S_{tot}) for the synchronous and asynchronous requests. The link buffer busy percentage (U_b) is then:

$$U_b = \frac{R_a \times S_{\text{tot}}}{\text{No of buffers}}$$

RMF Input: In Figure 32 on page 129 there are two systems, K11 and K12, sharing the same links. We have to calculate the total number of requests for the synchronous and asynchronous requests **i** summed from both systems, and we calculate the weighted average service time from the service times on the report (on the right of **i**).

Example: We assume that we have a CEC issuing 2222 CF requests/sec, and that the weighted average service time for those requests is 400 µs.

We assumed two links from the CEC to the CF, each link with two link buffers, giving four link buffers in all. When a synchronous (lock-type) request arrives, it is assigned to one of the link buffers. If there are no free buffers, the request is assigned to one of the buffers and it spins there until that buffer becomes free. This spinning request is not transferred to any other buffer, even though another buffer could become free. For that reason, we reduce the effective number of buffers in our calculation from four to three.

$$U_b = \frac{2222 \text{ req/s} \times 400\,\mu\text{s/req}}{3} = 0.30$$

The buffer is 30% busy.

Example: We have made calculations with three buffers. Suppose we use k=2. Then we have:

$$T_a = 2 \times \frac{0.30^3}{1 - 0.30^3} \times 355 = 20\mu s$$

This value should be close to the weighted average from Figure 32 on page 129 in the column `Delayed Requests, Avg Time, /All`.

The total time is:

$$\text{Total time} = 20\,\mu s + 355\,\mu s = 375\,\mu s$$

When you are trying to estimate the effect of changing the number of buffers (number of links), be careful. This model probably does not give the correct answer for your situation. If you reduce the queue time in the front of the link

buffers, you may just move that queue time to the front of the CF CP. Note, however, that this is *not observed in the model*.

5.2.4 Time Distributions and the k Factor

We can select the *k* factor based on the standard deviation to average ratio of the service time from RMF reports. And we can justify it further when calibrating the model to fit the reported service time.

The *k* factor is based on the inter-arrival and service time distribution coefficients, as follows:

$$k = \frac{C_a^2 + C_s^2}{2}$$

where:

k = factor from Allen-Cunneen G/G/C approximation formula
C_a^2 = coefficient for inter-arrival time distribution
C_s^2 = coefficient for service time distribution

A distribution coefficient is calculated as follows:

$$C^2 = \frac{(\text{Standard deviation})^2}{(\text{Average})^2}$$

You see two typical inter-arrival time distributions in Figure 46. One has a coefficient (C^2) of one (exponential distribution), and the other has a coefficient (C^2) of five (hyperexponential distribution). Additionally, we have arrivals with constant inter-arrival times, where the coefficient is zero, which means no variations in inter-arrival time. All of the distributions shown have an average inter-arrival time of 1000 units.

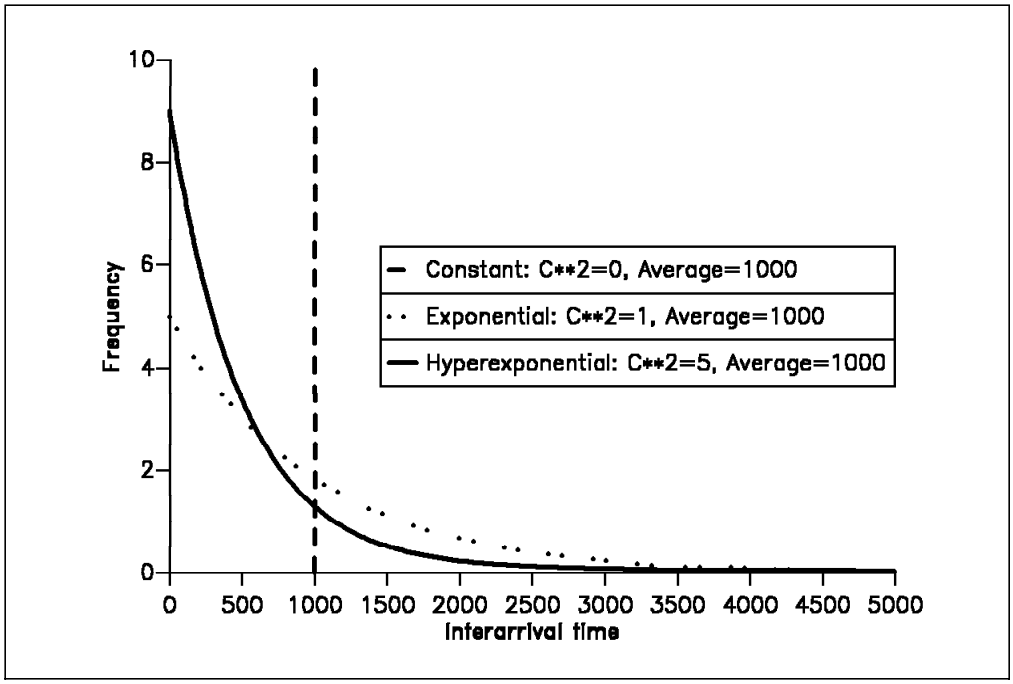


Figure 46. Typical Inter-arrival Time Distributions

In Figure 47 on page 155 you see two service time distributions. The one on the left represents lock requests and their service time distribution. Note that the service time distribution is close to constant since the service time coefficient is close to zero (0.1).

On the right you see service time distributions for all the asynchronous CF requests. Note that the service time distribution here can be approximated with an exponential service time distribution (the service time distribution is close to one).

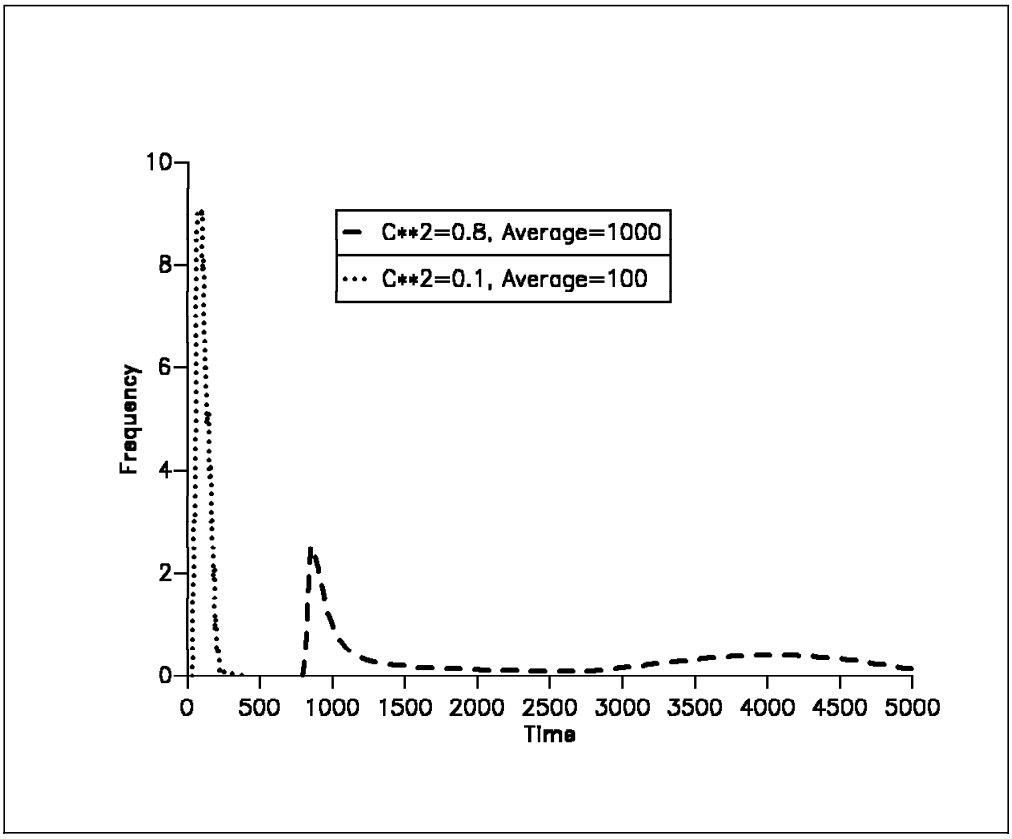


Figure 47. Two Separate Service Time Distributions

Let us figure out what happens if we combine the two service time distributions shown in Figure 47. In the (hypothetical) case where our server (CF) has unlimited capacity, the distributions will not affect each other. Figure 48 on page 156 shows the combined distribution.

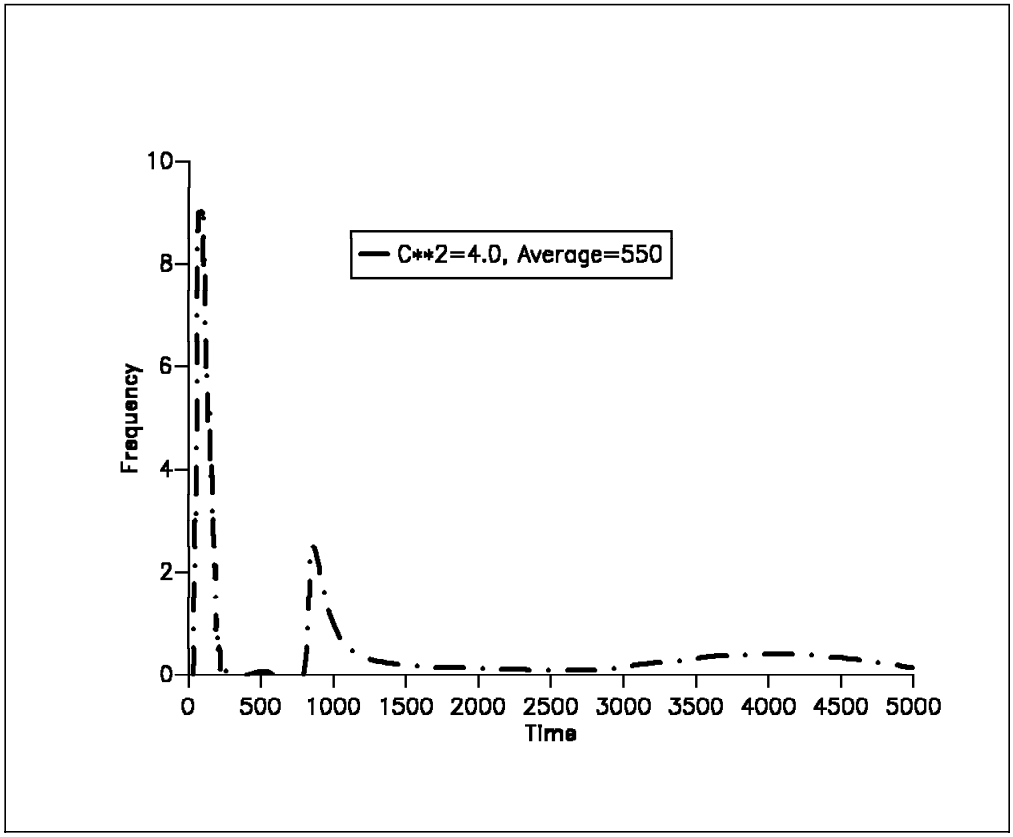


Figure 48. Combined Service Time Distribution (Unlimited CF Capacity)

Note that the coefficient (C^2) is now about four, and in this case the average service time is 550. The average does not have to be the mean of the service times shown in Figure 47 on page 155. The reason that the average is so easy to calculate is that the area under each curve is similar. However this is seldom the case in real life. The service time coefficient of four suggests a hyperexponential distribution (even though the distribution is not an exponential distribution).

Figure 49 on page 157 depicts the case where our server (CF) has limited capacity. Note that the average service time increases and that the peaks flatten out somewhat. We also observe that the service time coefficient decreases somewhat (from four to three).

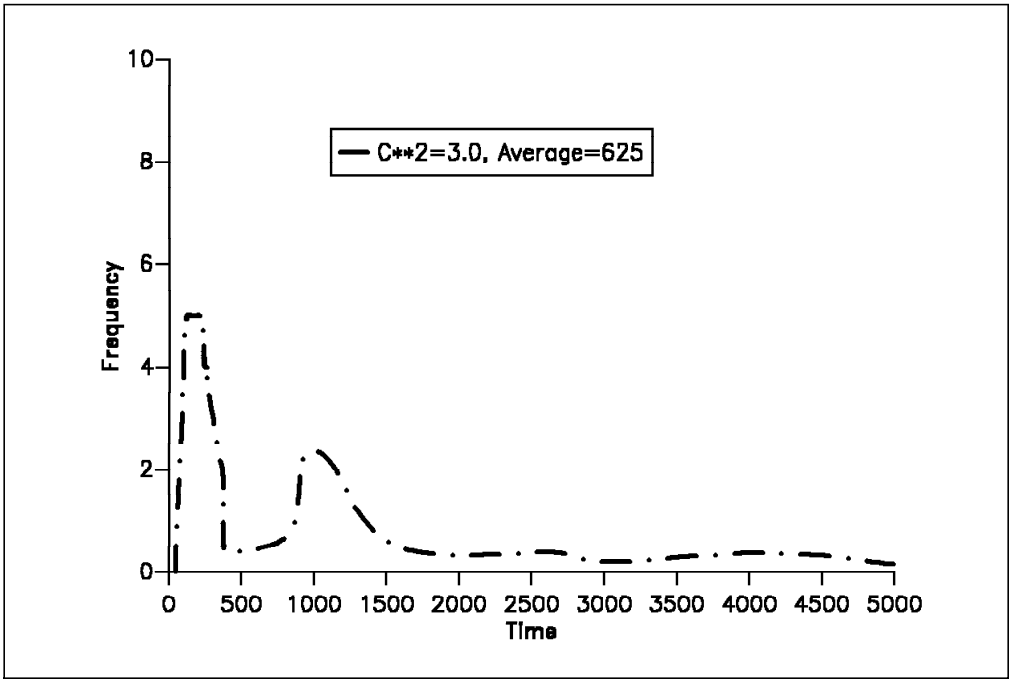


Figure 49. Combined Service Time Distribution (Limited CF Capacity)

5.2.5 Verifying the Model

The results of this model were compared to a few cases from some production environments.

Case 1: Case 1 has two CECs and four OS/390 systems connected to two CFs. Both CFs have one CP. The CECs are connected to the CFs using two links each. The subsystems are CICS and DB2. Application data sharing is 100%.

The first CF has a heavy but acceptable load. Against that, our model:

- Estimates 10% low for the total CF service time, when CF utilization is about 40% and the reported busy counts (expressed as delayed requests in % of the total number of issued requests) in RMF Subchannel Activity reports are about 3%.

The second CF has a very heavy load. Against that, our model:

- Estimates 10% low for the total CF service time, when CF utilization is about 58% and the reported busy counts (see note for the first CF) in RMF Subchannel Activity reports are about 300%.

See also 7.3, “Case Study: CF Modelling” on page 226.

5.2.6 RMF Reports

Let us relate our model to RMF reporting. Figure 50 on page 158 shows our model’s relationship to RMF time reporting.

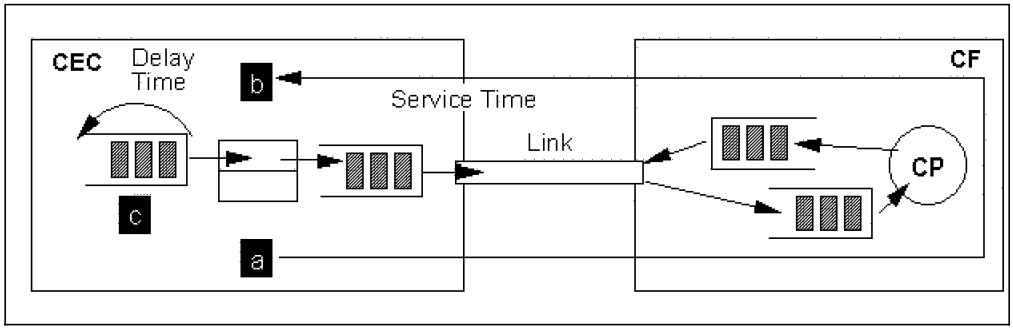


Figure 50. Relationship to RMF Reporting

When RMF reports the times of the CF requests, it splits the total time (response time) into *delay time* **c** and *service time* from **a** to **b**. Our model uses the same split. Additionally, our model further splits the service time (total service time) into three other service times and three queue times.

A piece of the delay time, when a synchronous request has gotten an available software subchannel but is spinning on the link buffer (only if sharing links on EMIF), is reported in RMF in *service time*. This piece of time is actually a queueing (delay) time, but it is taken as the service time from RMF reports in our model as well.

Refer to Chapter 4, “Using RMF Parallel Sysplex Reports” on page 117 for more information about RMF.

5.3 Using the CF Processing Model

To improve the performance of your CF configuration based on this model, start from the CF CPs as follows:

1. Begin with the CF CP queue time calculations. Try to decrease the queue time, for example, to less than one-third of the CF CP service time.
2. Then calculate the link times. Observe the effect of the link speed, HiPerLinks, and the number of links.
3. Calculate or assume the CEC service time. Observe the effect of HiPerLinks. Now you have the total CF service time.
4. Finally, calculate the link buffer queue time. Verify that this is reasonable. Observe the effect of the number of link buffers (the number of links).

When using this model, note the following:

- If the CF CP queue time is substantial, that is, more than one-third of the CF CP service time, then increasing the number of link buffers may lead to the wrong conclusion. In reality, the queued requests are only moved to the CF CP queue, but our model *does not observe this*.
- If the sending OS/390 systems share a CEC using PR/SM and use shared CPs, then the RMF service time for the immediate synchronous requests only includes the physical (real) spinning time of the CP. It does not include the possible time when some other OS/390 system is using the CP, when the PTFs for APAR OW12415 are implemented. Our model assumes wall clock time, not the physical CP time.

Finally one might ask: what do we gain by improving the total time of a CF request? Table 13 on page 159 answers this question with a series of examples. It assumes that one knows the value of one consumed CEC CP percent. Table 14 on page 159 gives examples of the CF response time improvements when changing the CF.

Additionally, observe that by improving the CF request response time, the application transaction response time could be improved.

Table 13. Examples of CEC CP Consumption Per Synchronous CF Requests

CEC Model	Requests per Second	Response Time Improvement (μs)	CEC % Consumed
R61	600	100	1
R83	1400	60	1
RX4	3000	33	1
R95	4000	20	1
982	4000	20	1

Table 14. Examples of Synchronous CF Request Response Time Improvements

From CF Model	To CF Model	CF Request Response Time Improvement (μs)
C01	C02	10-15
C02	C04	25-45
C02	C05	40-70

For a detailed step-by-step discussion, refer to 7.3, “Case Study: CF Modelling” on page 226 where an example of the use of the KGTV model is outlined.

Chapter 6. Using CP2000 for Capacity Planning in a Parallel Sysplex

At a Glance

This chapter introduces the CP2000 Quick-Sizer, an IBM capacity planning tool. We discuss the Quick-Sizer assumptions when migrating to Parallel Sysplex.

Recommended Reading:

- *CP2000 S/390 Parallel Sysplex Quick-Sizer User's Guide*
- *CP2000 Metrics Report for Sample Things*
- *CP2000 Performance and Capacity Report for Migration Analysis*

Up-to-date information is available from your IBM representative, who can reach the information on CPSTOOLS.

© Copyright IBM Corp. 1998

6.1 Capacity Planning with CP2000

Although CP2000 was still under development at the time of writing, in this section we discuss the IBM family of capacity planning tools that are based on CP2000.

CP2KEXTR (CP2000 Extract) extracts basic data from SMF and RMF and feeds CP2000 and DASD Magic. Figure 51 shows DASD Magic, which can be used to provide high-level capacity models for DASD (there is a similar facility for tape storage subsystems) and CP2000 for planning processor capacity. Note that CP2KEXTR runs on an OS/390 system. Further details on DASD Magic and CP2000 output to Freelance or WordPro are beyound the scope of this redbook.

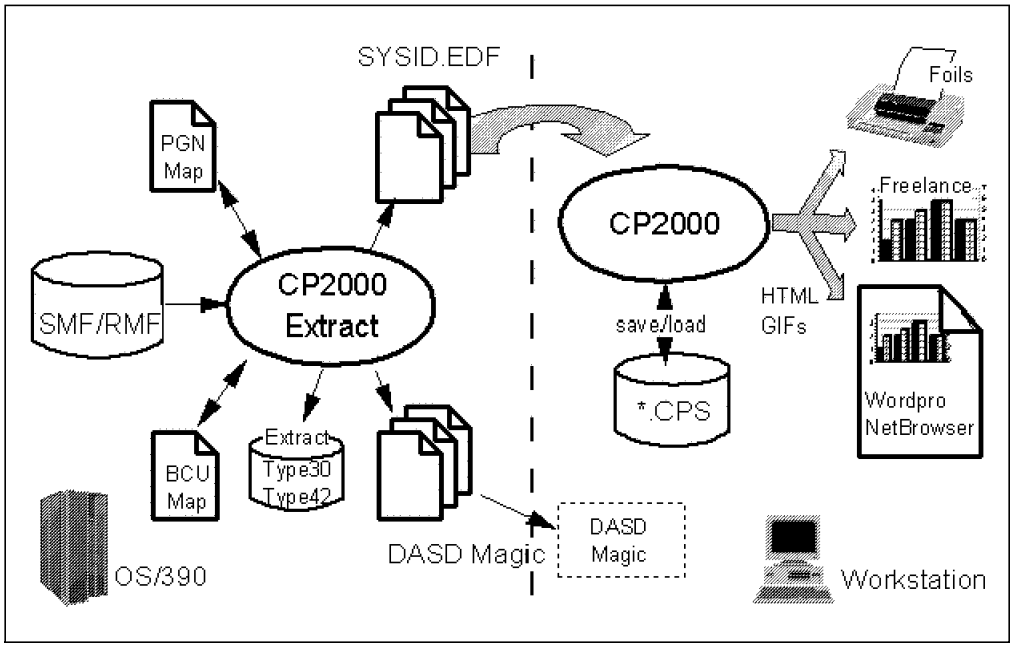


Figure 51. Overview of CP2000

6.1.1 Outline of CP2000 Usage

The following is a summary of the steps involved when using CP2KEXTR with CP2000, shown in Figure 52 on page 163.

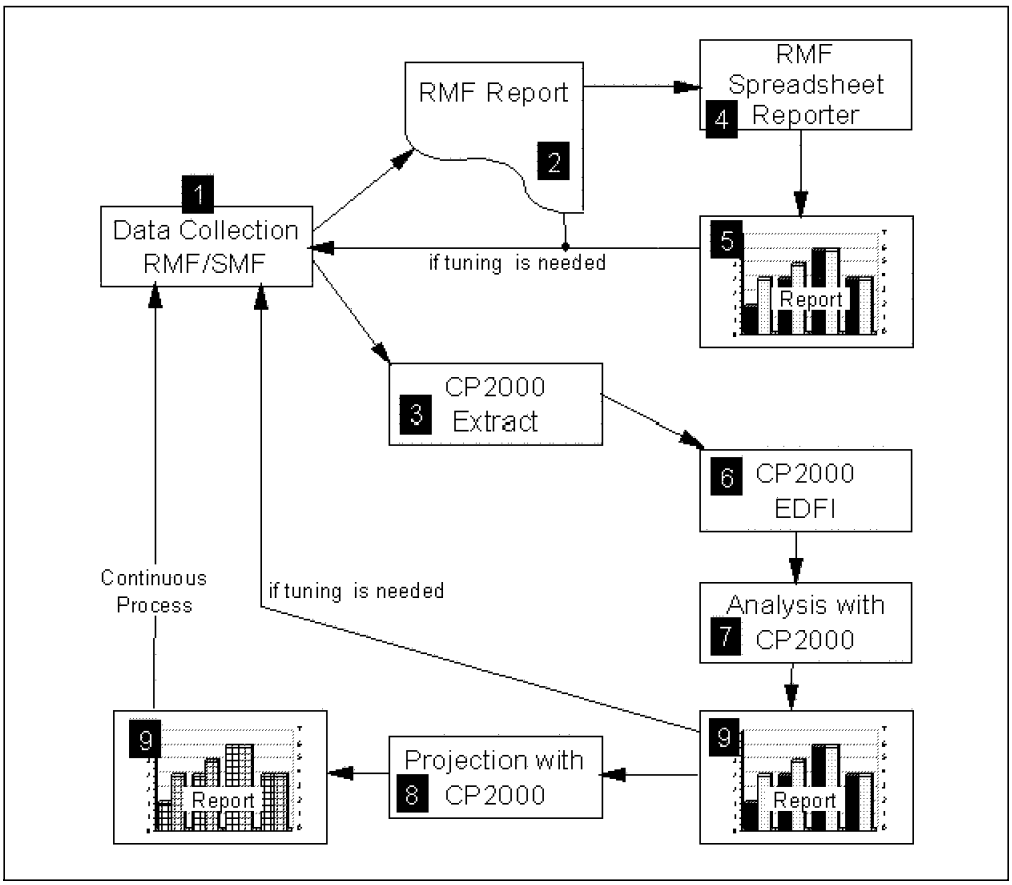


Figure 52. Capacity Planning Roadmap

1 Data collection SMF

Make sure you have data from representative periods. After SMF data collection, two alternatives are available: the data can be post-processed via either RMF or CP2KEXTR. *For a time-saving shortcut, the CP2KEXTR alternative is recommended.*

2 RMF report

Produce the RMF reports. When you produce the reports in machine-readable format, you can input them to the *Spreadsheet Reporter.*

3 CP2KEXTR

You select various SMF (type 70 to 79) records that are used by the CP2KEXTR program, which produces an *Enterprise Data File* (EDFI).

CP2KEXTR can also read SMF type 30 (job records), type 42.6 (detailed I/O statistics), and type 245 (CRR records). This provides a means of extracting CRR data for the IBM DMAGIC2 tool and providing input to the Comparison tool in CP2000.

4 RMF Spreadsheet Reporter

You can use the Spreadsheet Reporter to check for bottlenecks.

5 Report

If need be, tune the system to remove bottlenecks before carrying out capacity planning.

6 CP2000 EDFI

Make sure you have an EDFI from CP2KEXTR. Load this file into CP2000. It is possible to use CP2000 without this file, but then you have to create the numbers yourself from the reports. Manually inputting data to CP2000 can be time-consuming and error-prone.

7 Analysis with CP2000

At this point, make certain you have the right assumptions, profiles, current analysis, and so forth.

8 Projection with CP2000

Use all the future growth capabilities, such as different workload scenarios, different CEC alternatives, and so forth. Refer to 6.4.1, "Workload and CPU Projections" on page 191 for a short description of some of the CP2000 capabilities in this area.

9 Report.

CP2000 generates Performance and Capacity Reports (similar to display and describe reports in CP90).

Try It Out!

You are encouraged to try CP2000. There is a CP2000 case study that will take you through some of the steps just listed. *You do not have to provide your own data*, since it is an integral part of the case study. Refer to Chapter 7, "Parallel Sysplex Capacity Planning Case Studies" on page 195.

6.2 CP2000 Quick-Sizer Overview

The CP2000 Quick-Sizer is a productivity tool designed for use by IBM personnel. It is intended to provide insight into the capacity of IBM CECs when used in a Parallel Sysplex environment.

Quick-Sizer Input/Output

Minimal input is required for Quick-Sizer. Input describes current workloads, and that portion of the workloads targeted for Parallel Sysplex implementation. The results provide a high-level estimate for the configuration that would be required to support the designated workloads, including:

- Number and type of CFs
- CF storage sizes on a per structure basis
- Number of Coupling Facility links
- Number and type of CECs including 9672s
- CPU and Link utilizations

A Parallel Sysplex configuration containing 9021 711-based and 9021 511-based CECs can also be estimated by Quick-Sizer. In this case, you select the 9021 711-based or 9021 511-based CEC as the "current" CEC. The workload on the current CEC can be moved or migrated to a Parallel Sysplex containing both the current CEC and 9672-based CECs. Workload elements, whether they are eligible for Parallel Sysplex or not, can either stay on the current CEC or move to 9672-based CECs.

Quick-Sizer Panels

Figure 53 on page 165 displays the structure of the Quick-Sizer. After you have collected your input data **a**, you can start the tool, define your current environment (**b** and **c**), and examine the results on a set of output windows (**e**, **f** and **g**).

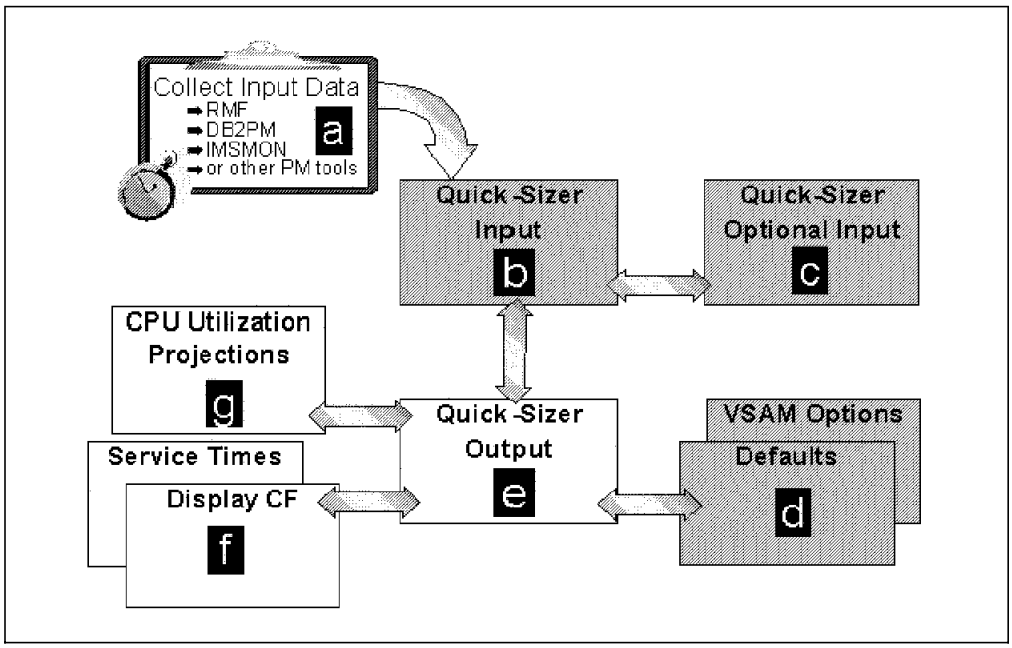


Figure 53. Quick-Sizer Structure

The Quick-Sizer assumes a set of default values used to project the current workloads from their single system environments to a data sharing environment in a Parallel Sysplex. These defaults can be changed on the Quick-Sizer Defaults and VSAM Options windows **d**.

Quick-Sizer Accuracy

The Quick-Sizer is accurate to within about 20%, so some caution should be applied in using its output if Coupling Facilities are near the recommended load limits. This is particularly the case when we are estimating CF capacity but certain input data may not (and cannot) be known accurately. For example, see 2.2.4, “DB2 Locking Rate Calculations” on page 67 for a discussion of DB2 locking rates, which must be estimated in many cases.

6.2.1 Defining Your Environment to Quick-Sizer

In this section, some important Quick-Sizer panels, as in Figure 53. are explained. The panels are annotated (**1**, **2**, and **A**, **B** ...) to facilitate the association between particular sections in the panels and their descriptions in the text.

An understanding of the input and output fields is essential for using Quick-Sizer.

Sample Scenario

To demonstrate Quick-Sizer, we use a sample scenario. It uses a 9021-821 as the current system. We assume that the 9021-821 has a total CPU utilization of 90%, and that it runs an IMS workload. Some of the workload is moved to one or more 9672 systems, and a portion of it remains on the current system and participates in Parallel Sysplex. The Quick-Sizer is used to calculate the scenario.

6.2.1.1 Quick-Sizer Input Definitions

Quick-Sizer Input Window

When you start the Quick-Sizer, you get the window shown in Figure 54. You can define up to four systems from which you want to move some workloads to a Parallel Sysplex environment. Each system is defined in an Input group box **A**.

The first Quick-Sizer input window provides the capability to define two systems. Two more systems can be specified by pressing **Add'l workloads** **C** at the bottom of the window.

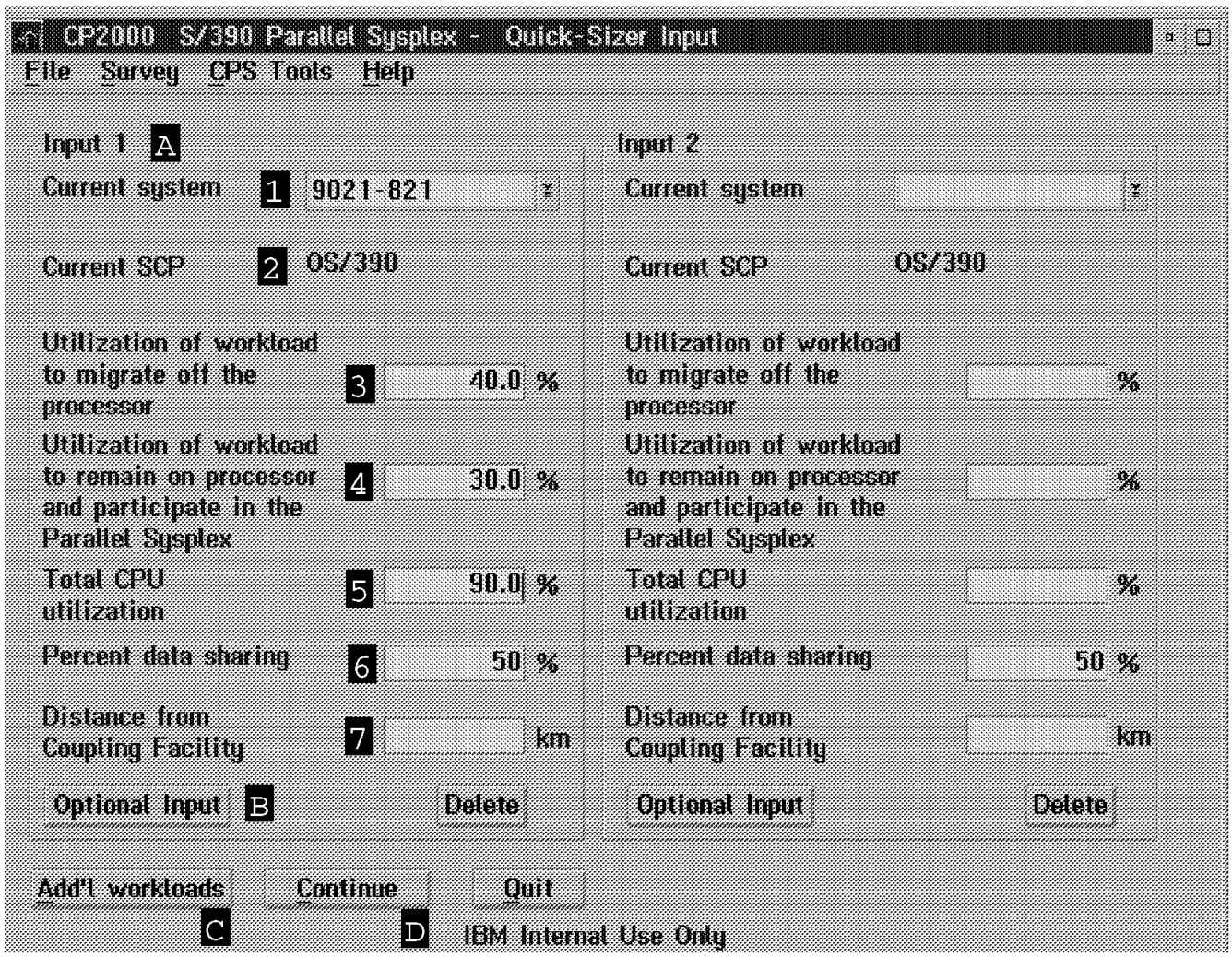


Figure 54. Quick-Sizer Input Window

Input Fields

1 Current system

This field denotes a CEC or CECs from which workload migrates to the Parallel Sysplex environment.

2 Current SCP

System Control Program (SCP) is the operating system on the current CEC. The Quick-Sizer accepts only:

OS/390 (OS/390 1.1, 1.2, 1.3 and 2.4)

Different M-values are used, based on the input. Projection is always to OS/390.

3 Utilization of workload to migrate off the processor

This is the CPU utilization in the current system of the workload elements to be moved to a 9672 configuration. This utilization represents the workload components that are *eligible to participate in Parallel Sysplex* and may include data sharing and non-data sharing workloads.

The CPU time for the workload elements that are moved must include all the CPU time these workload elements consume including MVS, VTAM, RACF, and so forth. Calculate this CPU time by applying *application capture ratios.* For a description on how to do this to your workload element, refer to 1.3.2, "OS/390 CPU Time Account and Capture Ratio" on page 14.

This workload element corresponds to **A** in Figure 60 on page 181.

4 Utilization of workload to remain on processor and participate in the Parallel Sysplex

Here you specify the CPU utilization of workload elements running on the current system that *remain there*. This utilization represents the workload components that are eligible to participate in Parallel Sysplex and may include data sharing and non-data sharing workloads.

The CPU time for the workload elements again must include all the CPU time these workload elements consume including MVS, VTAM, RACF, and so forth, as discussed for **3**.

This workload element corresponds to **B** in Figure 60 on page 181.

5 Total CPU Utilization

Here you specify the total CPU utilization of the current system. This utilization represents the sum of *all* workload elements.

To derive the workload elements that remain on the current system and are *not eligible* to participate in Parallel Sysplex, you may use the formula:

```
  Total CPU Utilization ( 5 )
minus Utilization of workload to remain on processor
  and participate in the Parallel Sysplex ( 4 )
minus Utilization of workload to migrate off the processor ( 3 )
```

This workload element corresponds to **C** in Figure 60 on page 181.

6 Percent data sharing

Here you may specify the percent of the current workload eligible for Parallel Sysplex that will do Parallel Sysplex data sharing. Notice that the percentage is the same for the current system and the 9672s.

This is illustrated in Figure 60 on page 181, and is discussed in more detail in 6.2.2, “Quick-Sizer Parallel Sysplex Migration” on page 180.

The workload elements that participate in Parallel Sysplex data sharing are labelled **B1** (current system) and **A1** (9672s). The corresponding workload elements that do not participate in Parallel Sysplex data sharing are labelled **B2** (current system) and **A2** (9672s).

The default Quick-Sizer value for the field “Percent data sharing” is 50%.

7 Distance from Coupling Facility

Here you may specify the distance of your current CEC from the CF. Utilization increases when the distance from the CF increases. This field is used when workload that remains on the CEC participates in the Parallel Sysplex and the CF is located in another location.

The distance between 9672s and the CF is described in the input field “Distance of 9672s from the Coupling Facility” (see **12** in Figure 56 on page 171). The maximum distance for multimode coupling links is 1 km. The maximum distance for single mode links is 3 km. Any distances above 3 km between a CEC and a CF require a Request For Price Quotation (RPQ).

For every km, 10 microseconds are added as propagation delay. Refer to Table 6 on page 61 for an overview of ITR degradations for the sending CEC.

Quick-Sizer accepts input of up to 20 km, and the default value is 0 km.

After defining the input you should press **Optional Input** **B** to define the workload to migrate to Parallel Sysplex, and the workload that remains on the current system and participates in the Parallel Sysplex.

Quick-Sizer Optional Input

The optional input window is shown in Figure 55 on page 169. This window has two sections, one for workload being migrated to Parallel Sysplex **E**, and one for any workload that is to remain on the current system while participating in Parallel Sysplex **F**.

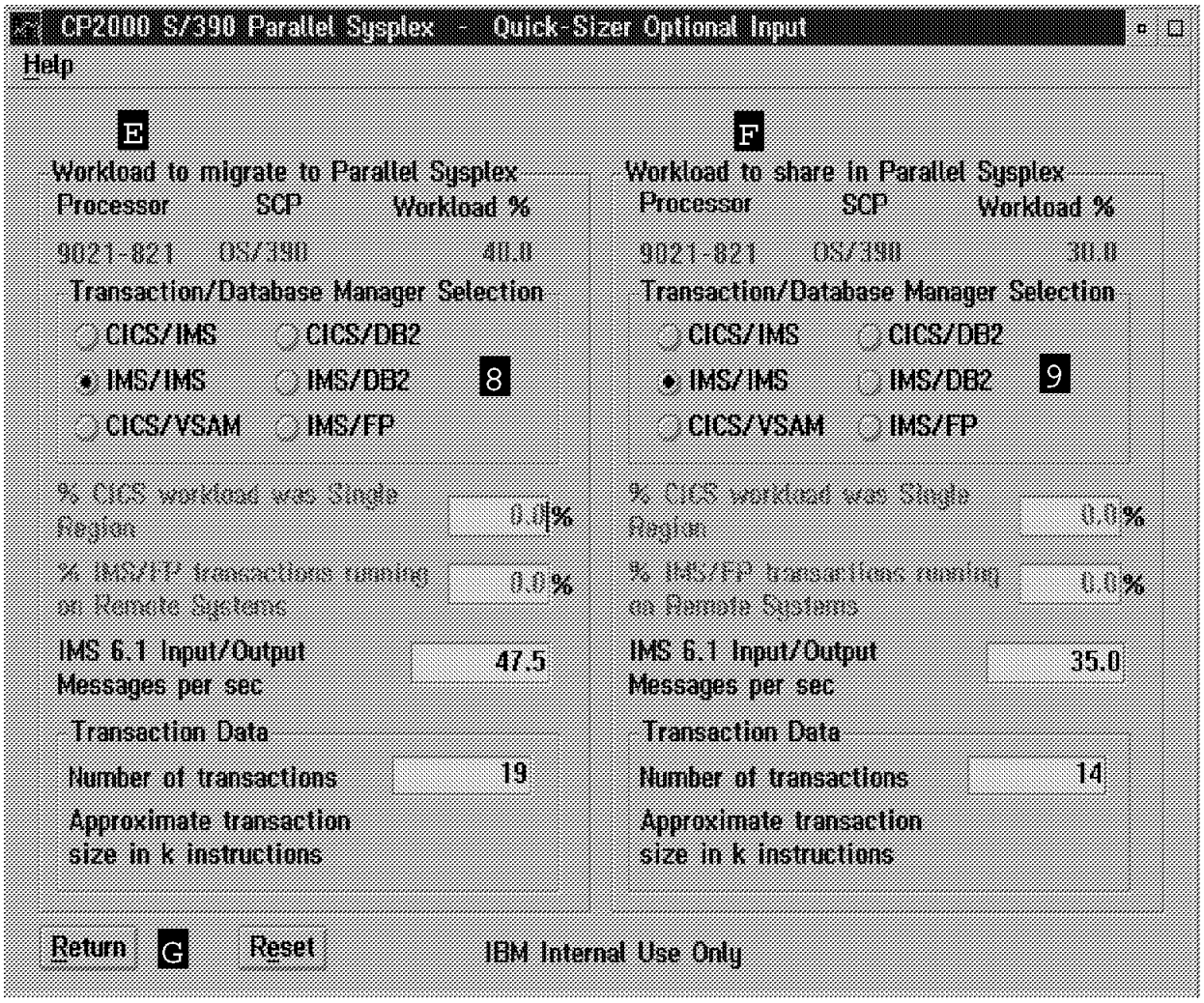


Figure 55. Quick-Sizer Optional Input Window

You can define six different transaction/database managers with Quick-Sizer. Click on the transaction/database combination that represents the workload. Quick-Sizer assumes **CICS/IMS** as default transaction/database manager. You can only define one transaction/database manager combination for the workload being migrated off the processor, and one combination for the workload that remains on the processor and participates in the Parallel Sysplex.

Planning for multiple transaction/DB managers

If you have to plan for different transaction/database manager combinations, you have to run Quick-Sizer for each combination and add the storage sizing and CPU projections manually. The one exception is if the different combinations are being migrated from the same processor. See page 174 for details of the Same Shared Processor option, which provides an exception to this rule.

Table 15 on page 170 shows the input fields applicable for each transaction manager.

Table 15. Input for Transaction/Database Manager Selection						
	Transaction/Database Manager					
Input Field	CICS/IMS	CICS/DB2	CICS/VSAM	IMS/IMS	IMS/DB2	IMS/FP
% CICS workload was Single Region.	√	√	√			
% IMS/FP transactions running on Remote Systems						√
IMS 6.1 Shared Message Queue messages per second				√	√	

If you click on any of these options, additional costs will be added for that option.

1. MRO costs are added to CICS workloads that currently run as a single region. To obtain the benefits of a Parallel Sysplex, MRO must be used.
2. In IMS V6.1, Fast Path allows transactions to run on a "remote" system (via the CF, not MSC), and Quick-Sizer adds the cost of doing this for the specified proportion of messages (default 0%). The IMS systems use a Shared Expedited Message Handling Queue (SEMHQ) structure in the CF, but FP prefers to run the transaction on the "local" system if possible.
3. Additional costs are incurred for IMS 6.1 Shared Message Queue (SMQ). Use of IMS SMQ involves a significant number of CF accesses per message and should only be specified if you are sure you want to use it.

For each transaction/database manager combination, you can modify the number of transactions per second for this workload. Quick-Sizer provides default values based on your system definitions. When you modify the number of transactions, the **Approximate transaction size in k instructions** field is displayed with data. This metric can be used as a reality check to validate the input data.

You can return to the main input window by clicking **Return** **G** .

Input Complete

After completing the input definitions, you can press **Continue** **D** to calculate the projections based on your specifications.

6.2.1.2 Quick-Sizer Output

On the next panel click **Compute**.

The Quick-Sizer output window (see Figure 56 on page 171) displays the results of the workload migration to Parallel Sysplex based on default values.

If you change the default values, then on return to this panel **Compute** must be clicked again to cause recalculation based on the values you have just specified.

- CPU utilization, link utilization, and the number of links for the current system **H** . In this case the "uplift" on the work remaining on the 9021-821 is 3% of the 821, that is, 3/50=6%.

- Target system requirements for the workload migrated to 9672 **I** and **K**, and for the CF **J**. **K** refers to links on the 9672 CEC in the corresponding place in **I**.

Additional output windows can be obtained by pressing **Service time** and **Display CF** **L**. You can examine CPU projections by selecting **Graph** **M** from the File menu.

You can also specify additional input parameters, denoted by **10** to **16** in Figure 56. These input parameter specifications help you to easily determine the impact of certain hardware features and environmental specifics.

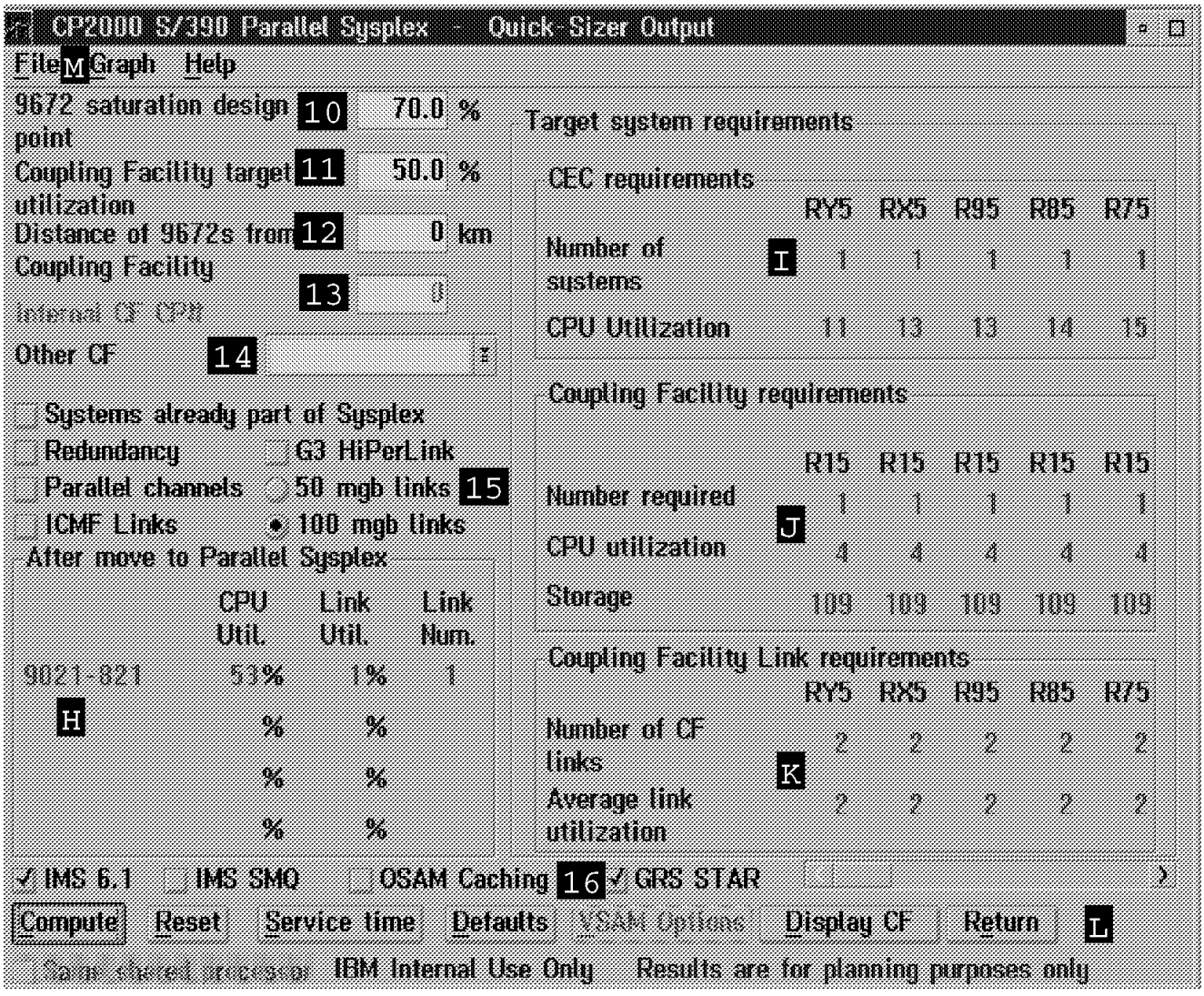


Figure 56. Quick-Sizer Output

Input Fields on Output Window

10 9672 saturation design point

The utilization calculated for 9672 is less than or equal to the saturation design point. The default value is 70%. Refer to 1.3.4, “Peak-to-Average Ratio (PAR) and Saturation Design Point (SDP)” on page 21 to get more information about the saturation design point.

11 Coupling Facility target utilization

The default target utilization is 50%. When configuring a CF, the utilization is less than or equal to the target specified. If the largest CF in the series exceeds target utilization, an additional CF is added. This value should be designated such that the peak period utilization is sustained without significant impact.

Note that if, as in our example, Redundancy is selected, the target utilization applies in the event redundancy is used. So if, in normal use, two CFs would run at, say, 30% busy, Quick-Sizer nevertheless configures three CFs. This is because if you had two CFs and one failed, then the surviving CF would run at around 60%, which is higher than our target utilization.

12 Distance of 9672s from Coupling Facility

Here you may specify the distance of the 9672s from the CF. The input field is different from **7** in Figure 54 on page 166, which defines the distance of the current processors to the CF.

13 Internal CF CP

This is the number of internal CF CPs. Specify if ICF uses a spare CP in a G3 or follow-on model.

14 Other CF

ICF can only be specified if this option is selected.

If a CEC is specified in this field, this CEC is used as the CF. The default is a 9674 belonging to the same family as the CEC (for example, R3-based CEC and CF). This selection criteria can cause the CF to be *over-configured*. Always apply a reality check to the configuration that Quick-Sizer proposes and if the selected CF is very low-utilized, consider choosing another model.

Imaginary 9674 Models

The 9674 models are indicated by, for example, 9674-R25, meaning a 2-way 9674-C05. The 9674-C04 models are indicated by Multiprise 2000 model numbers up to the 5-way (for example, the 9674-156 is a 5-way 9674-C04), and by 9674-Rx4 for the 6-10 way models (for example, the 9674-R64 is a 6-way 9674-C04).

For the CF specified, utilization is calculated and displayed to show processing power required to support the workloads migrated. LPAR management time in the CF is included in the calculation.

You may also enter as input 9674s with one to ten CPs.

15 This area contains the following input options:

- Systems already part of Sysplex

 If the CECs are already part of a sysplex, less cost will be incurred for multisystem management.

- Redundancy

 An additional CF is included in the configuration if redundancy is required. CF utilization, storage, and links are adjusted accordingly. For availability reasons, at least two CFs are recommended.

- Parallel Channels

The default assumes ESCON channels. If parallel channels are used in the configuration, certain limits exist as to the number of available coupling links.

- ICMF Links

 ICMF links may be selected if a CF is specified and is the same as one of the processors that runs a workload participating in the Parallel Sysplex.

- G3 HiPerLink

 HiPerLinks are optional for the G3 models and are always available for the G4 models. The check mark only affects G3 models and is ignored for other cases.

- 50 mgb/100 mgb links (50MB/100MB links)

 The default is the 100MB link mode.

16 This area contains the following input options:

- IMS 6.1

 For IMS/FP workloads, the check mark is always turned on because these workloads can only participate in a Parallel Sysplex for IMS6.1. For IMS full function, you can indicate whether IMS/ESA V5 or IMS/ESA V6 is being used. The check marks IMS SMQ and OSAM Caching can only be enabled if IMS/ESA V6 is used.

- IMS SMQ

 With IMS/ESA V6, IMS full function can use the CF for workload balancing and assured message delivery. If this option is used, all transactions will run using the shared message queue. A default transaction size of two and a half million instructions is used for the projection. The projection can be adjusted by doing the following:

 – Changing the number of transactions and/or the number of messages on the optional input window (see Figure 55 on page 169). The number of messages is generated using the number of transactions. When you change any of these numbers, the approximated size in k instructions message is displayed on the window to verify the changes.

 – Changing the IMS logger rate and/or the IMS shared message queue rate on the default window (see Figure 59 on page 178).

 Note: Quick-Sizer changed the default path length of one million to two and a half million instructions per transaction for IMS/IMS, CICS/IMS, IMS/DB2 and CICS/DB2 in August 1997. This leads to changed results when using default values compared to earlier versions of the tool, but is more representative of customer transactions and will make Quick-Sizer easier to use.

 IMS uses this shared message queue to add and remove messages for execution. Additional accesses to this structure also occur based on the amount of message queueing. It is possible to send asynchronous requests to the shared message queue if the message is greater than 4k. This is probably a very small number and should be included in the total rate entered.

 The projection is very dependent on correct specifications made on the optional input and default windows. You should check the input very carefully to get usable output.

- OSAM Caching

 With IMS/ESA V6, OSAM can use the CF cache structure to optionally hold changed data. This is done on a buffer pool basis. The projections can be adjusted by changing the IMS OSAM R/W cache ratio on the default window.

- GRS STAR

 With OS/390 R2 and follow-on releases, GRS can be used in star and ring mode. In star mode, GRS uses a CF lock structure that contains information for all global ENQ/DEQ requests. This reduces the traffic on XCF, because it is no longer necessary to send the GRS token to all systems participating in the Parallel Sysplex.

Same Shared Processor

The purpose of the **Same shared processor** option is to enable up to four different sysplex-enabled workloads to be migrated from one source processor. This is done by specifying the same processor type and model, with the same total utilization and with the same "Utilization of workload to remain on processor and participate in the Parallel Sysplex," on the initial input panel (Figure 54 on page 166 shows an example of the initial panel but not filled in for this option). In addition, CPU utilization of shared and migrated workloads cannot exceed the total CPU utilization.

Obviously, these specifications are all necessary conditions if Input 1 and Input 2 represent the same physical processor.
In this case, the **Same shared processor** button is enabled. Clicking it confirms this.

Compute Once Input Complete

Once the inputs have been set as desired, click the **Compute** button to calculate the overall Parallel Sysplex requirements. To restore the output window inputs to their default values, click the **Reset** button.

Output Fields

The output window Figure 56 on page 171 displays the result after migration to Parallel Sysplex:

H After move to Parallel Sysplex

The CPU utilization, link utilization and number of links are shown for the current systems, if they participate in Parallel Sysplex.

The CPU utilization is calculated by Quick-Sizer and includes the following workload elements:

- Elements remaining in the current system and eligible to participate in the Parallel Sysplex.

 This workload can further be subdivided into:

 – A workload element that is doing Parallel Sysplex data sharing
 – A workload element that is not doing Parallel Sysplex data sharing

- Elements remaining in the current system and not eligible to participate in the Parallel Sysplex.

These workload elements are pictured in Figure 60 on page 181 as **B1**, **B2**, and **C1**.

The link utilization shows the estimated utilization of the coupling links connecting the current system to the CFs (or CF if redundancy was not specified). It depends on the load, the distance, the speed, and the number of links. Quick-Sizer uses high speed coupling links (100 MB/sec) as the default.

The number of links describes the estimated number of coupling links connecting the current system to the CF. Quick-Sizer uses an algorithm to ensure that the probability of a CF request finding the link busy is small. It is estimated by Quick-Sizer such that the *combined total link busy* percentage is below a threshold of 10%.

Note: The combined total link busy percentage is a function of the *number of links*.

I CEC requirements

The CEC requirements show the number of 9672 systems required to run the workload migrated off the current processors (see **3** in Figure 54 on page 166). CPU Utilization displays the projected CPU utilization on the new systems. This amount is pictured in Figure 60 on page 181 as **A1** and **A2**.

Quick-Sizer offers a best choice, which is displayed in pink. In our example, Quick-Sizer suggests to use one 9672-RB5 for the workload that should be migrated to a 9672 system.

J Coupling Facility requirements

Here Quick-Sizer displays the number of required CFs, the projected CPU utilization, and the projected storage size for the CFs. In our example, Quick-Sizer suggests using two 9672-R15, because we wanted Redundancy **15**. A more detailed CF storage breakdown is shown in Figure 58 on page 177

K Coupling Facility link requirements

Quick-Sizer shows the number of CF links and the assumed link utilization for the CEC displayed in **I**.

Detail Output

Quick-Sizer calculates service time projections and detailed CF storage utilization by structure type. Use the buttons **Service time** and **Display CF** in button area **L** to obtain more detailed information. Additional graphics for CPU utilization can be examined by selecting **Graph** **M** on the file menu.

Service Time Analysis

Figure 57 on page 176 displays the CPU service time on the current system **N** and estimated CPU service time on the 9672s after migration. This helps you to determine the best configuration to meet your service time objectives. This type of analysis could be important for the early generations of CMOS. However, as the engine speed of CMOS now is similar to the engine speed of bipolar machines this analysis is becoming less of an issue.

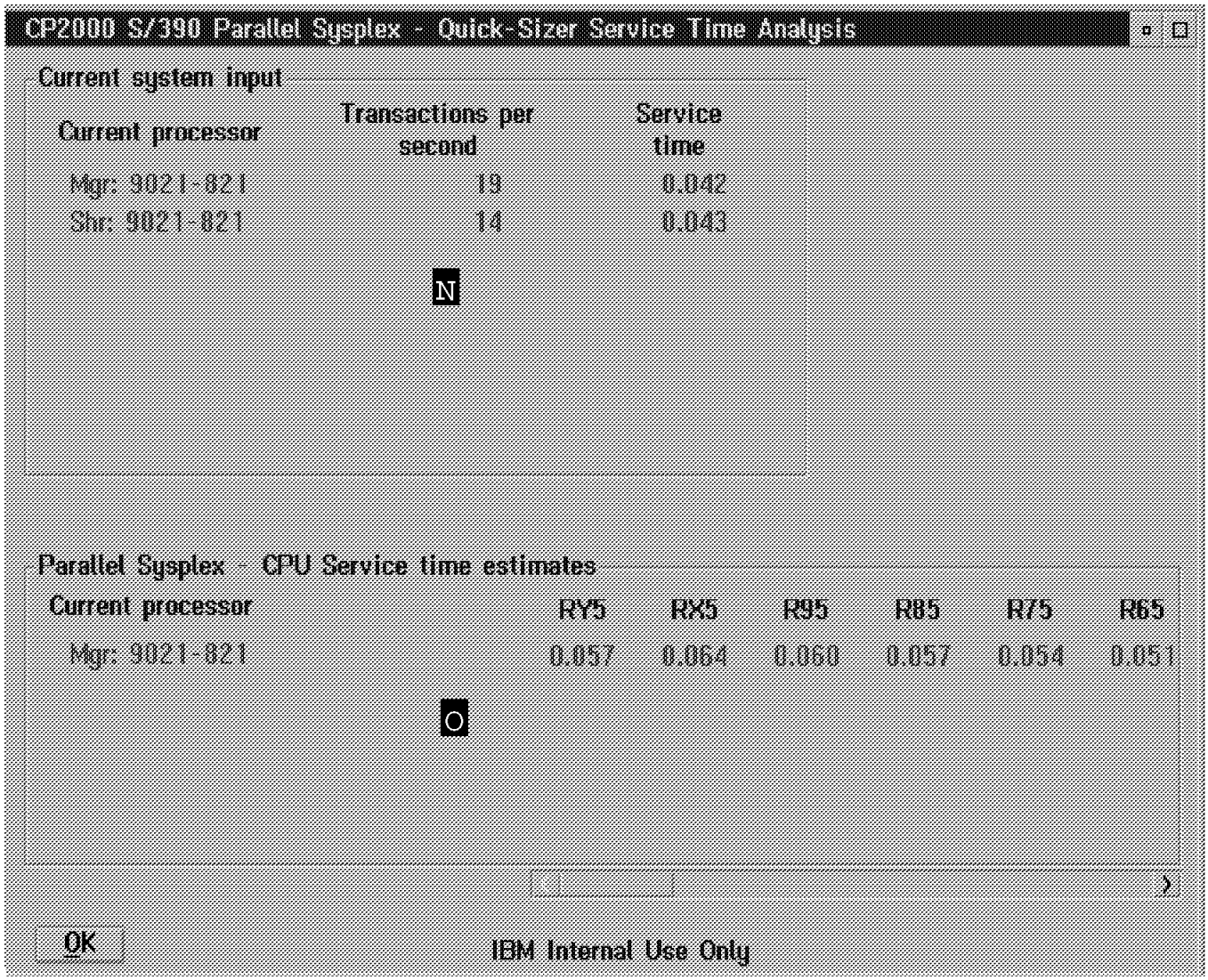


Figure 57. Quick-Sizer Service Time Analysis

Coupling Facility Structure Display

Figure 58 on page 177 helps you to determine the structure size for all structures in the CF **P**. You can change the Coupling Facility **14** and 9672 Configuration to your needs. When you change the CF, the input field Other CF **14** on the output window Figure 56 on page 171 is changed accordingly, and all calculations are based on the new CF.

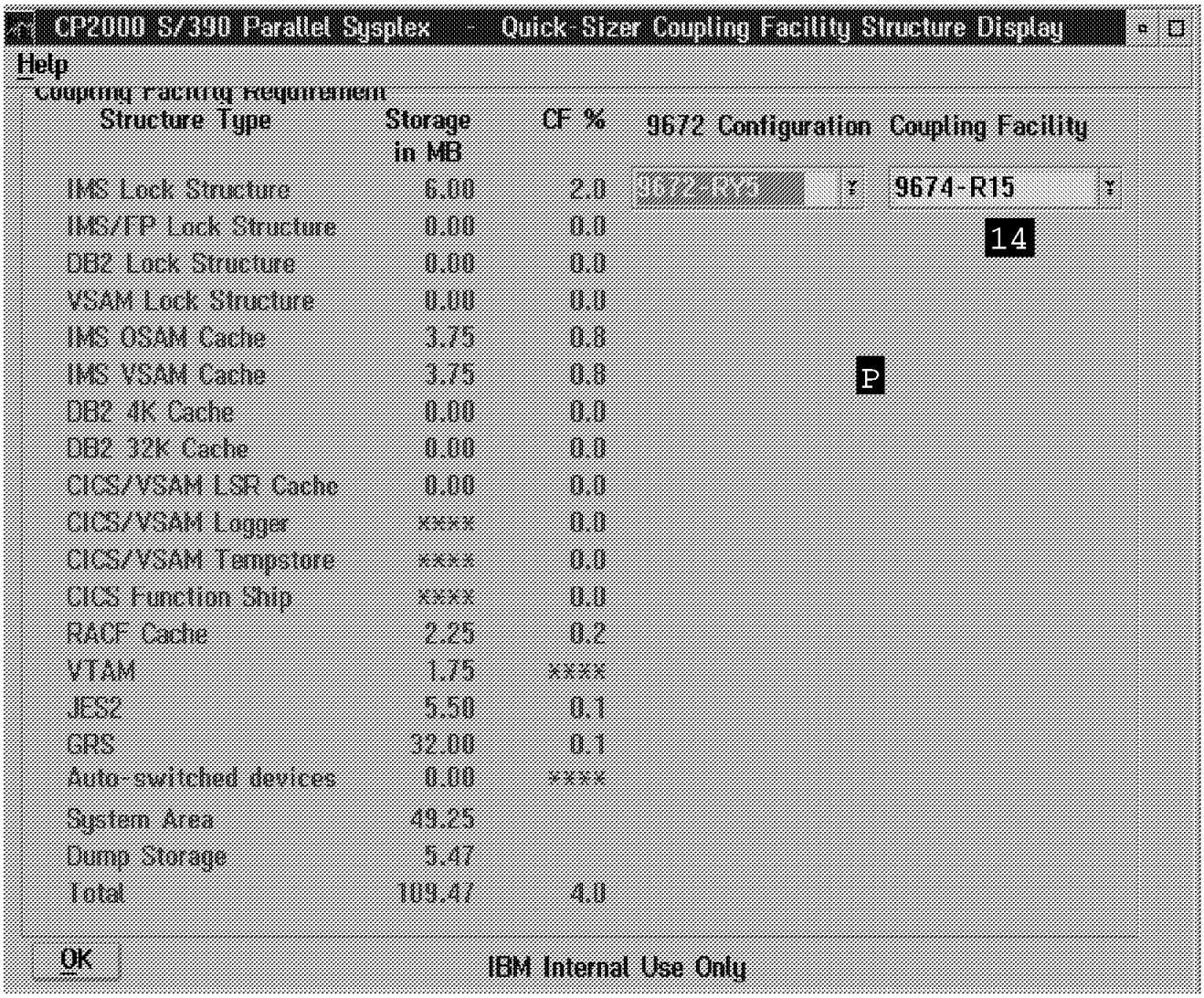


Figure 58. Quick-Sizer Coupling Facility Structure Display

6.2.1.3 Quick-Sizer Model Input

Quick-Sizer offers two dialogs to modify the model assumptions on which the calculations are based. Figure 59 on page 178 shows the Quick-Sizer Defaults window which contains the most important information, the Access Rate Calculation Input **R**. In addition, the defaults window helps you to adjust the storage calculation input **Q**.

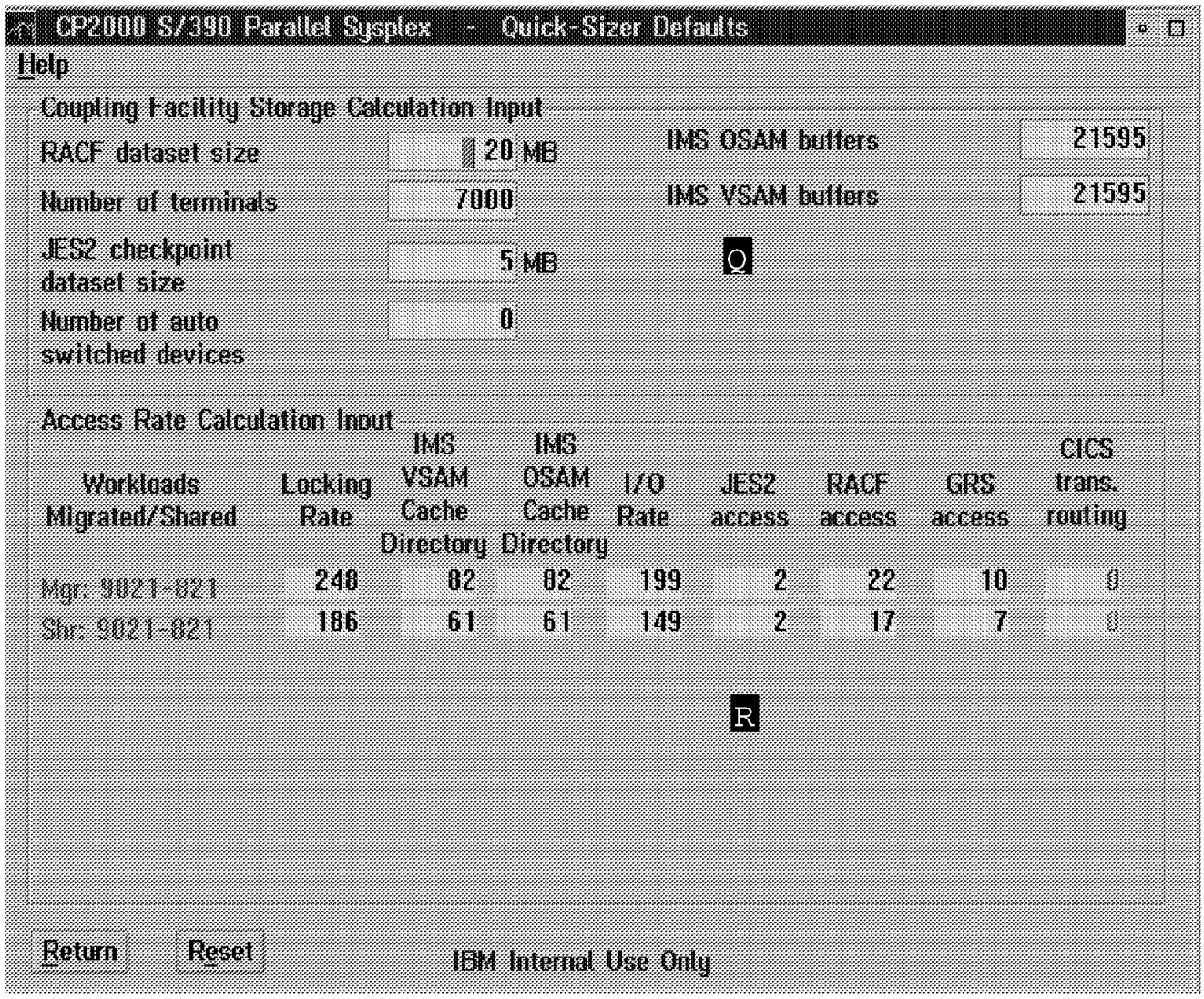


Figure 59. Quick-Sizer Defaults

Section 6.3, “CF Access Rate Calculation” on page 184 gives a more detailed overview on how to specify the access rates for IMS/DB2. Table 16 on page 179 shows which access rates are applicable for which transaction/database manager combination.

Table 16 (Page 1 of 2). Access Rate Input for Transaction/Database Managers

	CICS/IMS	CICS/DB2	CICS/VSAM	IMS/IMS	IMS/DB2	IMS/FP
General Access Rates						
Locking Rate	√	√	√	√	√	√
I/O Rate	√	√	√	√	√	√
JES2 Rate	√	√	√	√	√	√
RACF Rate	√	√	√	√	√	√
GRS Rate	√	√	√	√	√	√
CICS trans. routing	√	√	√			
DB2 Access Rates						
DB2 R/W w/o Data		√			√	
DB2 Read with Data		√			√	
DB2 Write with Data		√			√	
DB2 Page List Rate		√			√	
DB2 Batch Unlock		√			√	
DB2 32K R/W Rate		√			√	
CICS/VSAM RLS Access Rates						
VSAM Read w/o Data			√			
VSAM Read 4K with Data			√			
VSAM Write w/o Data			√			
VSAM Write 4k with Data			√			
VSAM Unlock Cast Out Lock			√			
VSAM Local Buffer			√			
CICS Logger R/W access			√			
CICS tempstore R/W Aux Sync			√			
CICS tempstore R/W Aux			√			
CICS tempstore R/W Main Sync			√			
CICS tempstore R/W Main			√			

Table 16 (Page 2 of 2). Access Rate Input for Transaction/Database Managers

	CICS/IMS	CICS/DB2	CICS/VSAM	IMS/IMS	IMS/DB2	IMS/FP
IMS OSAM, Fast Path and Shared Message Queue Access Rates						
IMS VSAM Cache Directory	√			√		√
IMS OSAM Cache Directory	√			√		√
IMS OSAM R/W Cache	•			•, ••		
IMS F/P VSO Read						√
IMS F/P VSO Write						√
IMS F/P Logger						√
IMS F/P Shared Queue						√
IMS Logger Sync				••		
IMS Logger Async				••		
IMS Shared Msg Queue				••		

Note:

- If IMS OSAM Cache is used (IMS/ESA V6)
- •• If IMS SMQ is used (IMS/ESA V6)

6.2.2 Quick-Sizer Parallel Sysplex Migration

Let us try to analyze the workload flow when you migrate to a Parallel Sysplex, as seen by Quick-Sizer, by looking at a sample scenario.

Figure 60 on page 181 shows an example that illustrates partial migration of workloads from a 9021-821 to two 9672s. The scenario is such that after the migration, both the 9021-821 and the 9672 constitute a Parallel Sysplex. Observe that some of the workloads remain on the 9021-821.

1 **2** **3** ... The annotated numbers refer to input or output fields on the Quick-Sizer panel, as discussed in 6.2.1.1, “Quick-Sizer Input Definitions” on page 166.

A **B** **B1** **B3** ... The annotated letters represent a subset of the total workload. Sometimes the input or output fields correspond to a workload element. Take a look at Figure 60 on page 181, which provides an overview of what Quick-Sizer can do for you.

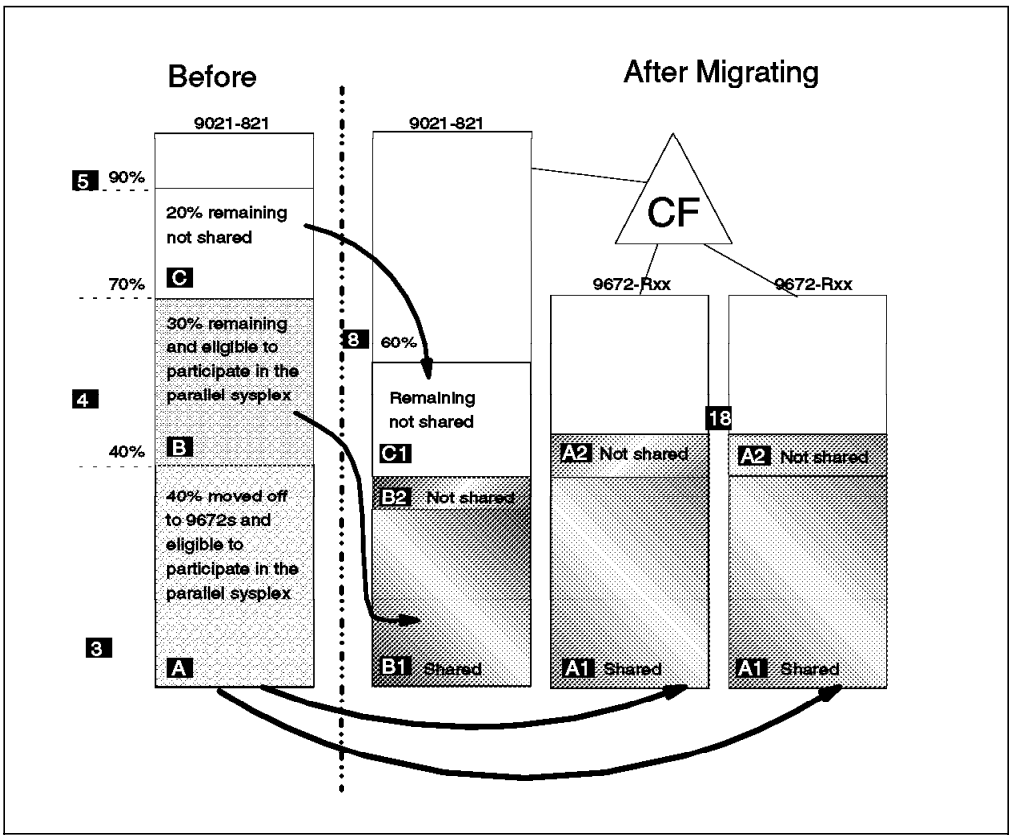


Figure 60. Parallel Sysplex Workload Migration - Overview

Note the following assumptions:

- The system on the left shows the original system prior to migration. Quick-Sizer lets you specify whether or not the original system was in a sysplex.
- There is one OS/390 system per CEC.
- All the percentages shown in the figure are CPU utilizations normalized to the respective CEC in which they appear.

6.2.2.1 Single 9021-821 System

9021-821 Workload Composition Before Migration

In the “before” situation, shown on the left side of Figure 60, there is no Parallel Sysplex data sharing. All the workload elements run in one OS/390 image, and the total CPU utilization is 90% on the 9021-821.

It is your responsibility to determine if a workload is eligible to participate in a Parallel Sysplex. When you move a workload from a single system to one or more 9672s, you describe your workload elements to CP2000 in very simple terms:

- Utilization of workload to migrate off the processor
- Utilization of workload to remain on the processor and participate in the Parallel Sysplex

For both the workloads above that participate in the Parallel Sysplex, you also specify the data-sharing percentage.

This percentage applies to installations who think they know what proportion of the updates are to shared data. Specify 100% unless you have a good reason to choose otherwise. Based on this information, Quick-Sizer calculates the overhead associated with the Parallel Sysplex, which is composed of both the *multisystems management cost* and the *data-sharing cost*. The data-sharing cost applies *only* to the percentage of the workload that does data sharing as specified in the `% data sharing` field. The Parallel Sysplex cost depends on the subsystem type, workload details, the CEC and CF type, and so forth. If the default values are used for the configuration, then the Parallel Sysplex cost is calculated in a manner equivalent to the formulas previously mentioned in:

2.2.3, “Parallel Sysplex CPU Cost for DB2 Data Sharing” on page 65
2.3.3, “Parallel Sysplex CPU Cost for IMS/DB Data Sharing” on page 77

The workload element in the field entitled Utilization of workload to migrate off the processor **3** (40% in the example) in Figure 60 on page 181 describes the percentage (in terms of the current CEC) of the workload moved from the 9021-821 to the 9672s. This workload element is eligible to participate in the Parallel Sysplex and part of it exploits Parallel Sysplex data sharing. You specify how much of this workload shares data in the field labelled Percent data sharing **6** (50% in the example).

In Figure 60 on page 181, data sharing is illustrated for the workload elements **A** on the 9021-821 and the corresponding **A1** (Shared) and **A2** (Not shared) on the 9672s.

The field labeled Utilization of workload to remain on processor and participate in the Parallel Sysplex **4** (30% in the example) in Figure 54 on page 166 describes the workload element eligible to participate in the Parallel Sysplex and *remains on the 9021-821.* This workload element is eligible to participate in the Parallel Sysplex and part of it exploits Parallel Sysplex data sharing. You describe how big of a part of this workload is data sharing in the column labelled Percent data sharing **6** (80% in the example). In Figure 60 on page 181, it corresponds to the workload element **B** on the 9021-821 and the corresponding **B1** and **B2**.

Part of the workload that currently runs on the 9021-821 is *not moved* to the 9672s and is *not eligible for participating in the Parallel Sysplex* (20% in the example). To derive this figure, Quick-Sizer subtracts the workload eligible to participate in the Parallel Sysplex from the current total 9021-821 CPU utilization, pictured at Total CPU Utilization **5** ((90-40-30)% = 20%) in this example). In Figure 60 on page 181, it corresponds to the workload element **C**.

Let us now proceed to the right side of Figure 60 on page 181 to find out what Quick-Sizer calculates.

6.2.2.2 Parallel Sysplex: 9021-821 and Two 9672s

9021-821 Workload Composition After Migration

Our 9021-821 shown in Figure 60 on page 181 now has three workload elements, **B1**, **B2**, and **C1**.

B1 denotes the workload element that is exploiting Parallel Sysplex functions, *including* Parallel Sysplex data sharing.

Its CPU utilization is 80% of 30% (9021-821 CPU utilization) plus the Parallel Sysplex cost *including* the multisystems management cost and the Parallel Sysplex data-sharing cost.

Quick-Sizer does not report this number explicitly.

B2 denotes the workload element that is exploiting Parallel Sysplex functions *excluding* Parallel Sysplex data sharing.

Its CPU utilization is 20% of 30% (9021-821 CPU utilization) plus the multisystems management cost.

Quick-Sizer does not report this number explicitly.

C1 denotes the workload element that remains on the 9021-821 and was not eligible to participate in the Parallel Sysplex.

Its CPU utilization remains almost unchanged and is 20% plus the multisystems management cost.

The previous topic explains how this number was derived.

The total CPU utilization of the 9021-821 is calculated by Quick-Sizer and is shown in “CPU Util after move.” (the actual Quick-Sizer output panel is not shown here). In Figure 60 on page 181, this is labeled **8**, and is 60% in our example.

The “uplift” in this particular case can be calculated as follows:

60% - (90% - 40%) = 3% or **8** - (**5** - **3**)

The relative uplift is 10/50%, giving 20%.

Recommended Parallel Sysplex Configuration

Quick-Sizer calculates the CPU utilization for a *range* of CEC/CF combinations. The recommended configuration is one having a CPU utilization close to but below your choice of *saturation design point* (SDP). The recommended configuration is clearly visible since it is shown in pink (based on our SDP). In our case we chose an SDP of 70%. The Quick-Sizer input field for the SDP appears on an output panel similar to Figure 56 on page 171 on which **10** denoted the SDP.

The aim of this discussion is not to show the recommended configuration, but rather to discuss some valuable features of Quick-Sizer. Let us therefore choose a configuration where the Parallel Sysplex comprises our 9021-821 and two 9672s.

9672 Workload Composition After Migration

The workload elements in the two 9672s are distributed evenly. In each 9672 we have two workload elements labelled **A1** and **A2**.

A1 denotes the workload element that is exploiting Parallel Sysplex functions including Parallel Sysplex data sharing.

Quick-Sizer does not report the CPU utilization of this workload explicitly.

A2 denotes the workload element that is exploiting Parallel Sysplex functions excluding Parallel Sysplex data sharing.

Quick-Sizer does not report the CPU utilization of this workload explicitly.

The total CPU utilization of each 9672 is calculated by Quick-Sizer and is shown in CPU utilization of 9672 systems. In Figure 56 on page 171, this is shown in column **I** for a range of configurations.

6.3 CF Access Rate Calculation

6.3.1, “DB2 CF Access Rate Calculation” provides information for the Parallel Sysplex DB2 environment. This discussion is based on Quick-Sizer.

For a detailed step-by-step discussion on how to do CF access rate modelling in a DB2 environment, refer to 7.1, “Case Study: DB2 Migration to a Data Sharing Environment” on page 197.

6.3.2, “IMS/DB CF Access Rate Calculation” on page 188 provides information for the Parallel Sysplex IMS/DB environment. This case study is based on Quick-Sizer.

6.3.1 DB2 CF Access Rate Calculation

Quick-Sizer provides DB2 Access Rate modelling. The modeled environment assumes DB2 V5 with Type 2 Indexes. The impact of DB2 V5 on data-sharing cost is expected to be small in many situations, but it is expected that there will be less variance in the locking rate distribution.

Locking Rate

The locking rate field is obtained by clicking **Defaults** on the output window (see Figure 56 on page 171) to obtain a panel similar to Figure 59 on page 178. The input required by Quick-Sizer for Group Buffer Pool access rates can be obtained from the average per-second column in the DB2PM Local Buffer Pool report. Full details can be found in 7.1.2.1, “Current Locking Rate Calculation (DB2PM)” on page 204. Quick-Sizer expects the locking rate to refer to Type 2 indexes, and so a scale factor should be used if you are not using them.

2.2.4, “DB2 Locking Rate Calculations” on page 67 contains full details.

As presented in 2.2.4, “DB2 Locking Rate Calculations” on page 67, the locking rate scaling factors are:

- $Lock\ Rate_{DB2V3} \approx 0.67 \times Lock\ Rate_{DB2V2}$
- $Lock\ Rate_{DB2V4T1x} \approx 1.00 \times Lock\ Rate_{DB2V3}$
- $Lock\ Rate_{DB2V4T2x} \approx (0.25\text{-}0.75) \times Lock\ Rate_{DB2V3}$

 The low end 0.25 of the range for DB2 V4 with Type 2 Index relates to a low R/W ratio. The high end 0.75 of the range for DB2 V4 with Type 2 Index relates to a high R/W ratio. IBM laboratory measurements show a 0.5 saving with an R/W ratio of 4.
- $Lock\ Rate_{DB2V5T2x} \approx (0.9\text{-}1.00) \times Lock\ Rate_{DB2V4T2x}$

Using the conservative assumption that you save half the lock/unlock requests when implementing Type 2 Indexes, we can make the following list for the migration value m:

m ≈ 0.67, if migrating from DB2 V2 to V3 or V4/5 with Index Type
m ≈ 0.34, if migrating from DB2 V2 to V4/5 with Index Type 2
m ≈ 1.0, if migrating from DB2 V3 to V4/5 with Index Type 1
m ≈ 0.5, if migrating from DB2 V3 to V4/5 with Index Type 2
m ≈ 0.5, if migrating from V4 Index Type 1 to Index Type 2
m ≈ 1.0, if migrating from V4 Index Type 2 to V5 Index Type 2

Other Fields

The first defaults panel also contains the I/O rate for the database. Use this number if you have it available, but the model is not very sensitive to it.

All the other fields directly concerned with DB2 data sharing are found by clicking the bottom right arrow on the first defaults panel (see Figure 59 on page 178).

An optional panel similar to Figure 61 on page 186 shows you the input fields for detailed DB2 CF Access Rate Calculations. And note that some of the input field headings on the Quick-Sizer panel do not always map directly to the headings in the DB2PM reports. Sample DB2PM reports are provided in 7.1, “Case Study: DB2 Migration to a Data Sharing Environment” on page 197. To get access to this window, you have to specify IMS/DB2 or CICS/DB2 in Optional Input in the initial input panel (see Figure 54 on page 166 indicator **B**). Then you can reach this window by selecting **Defaults.**

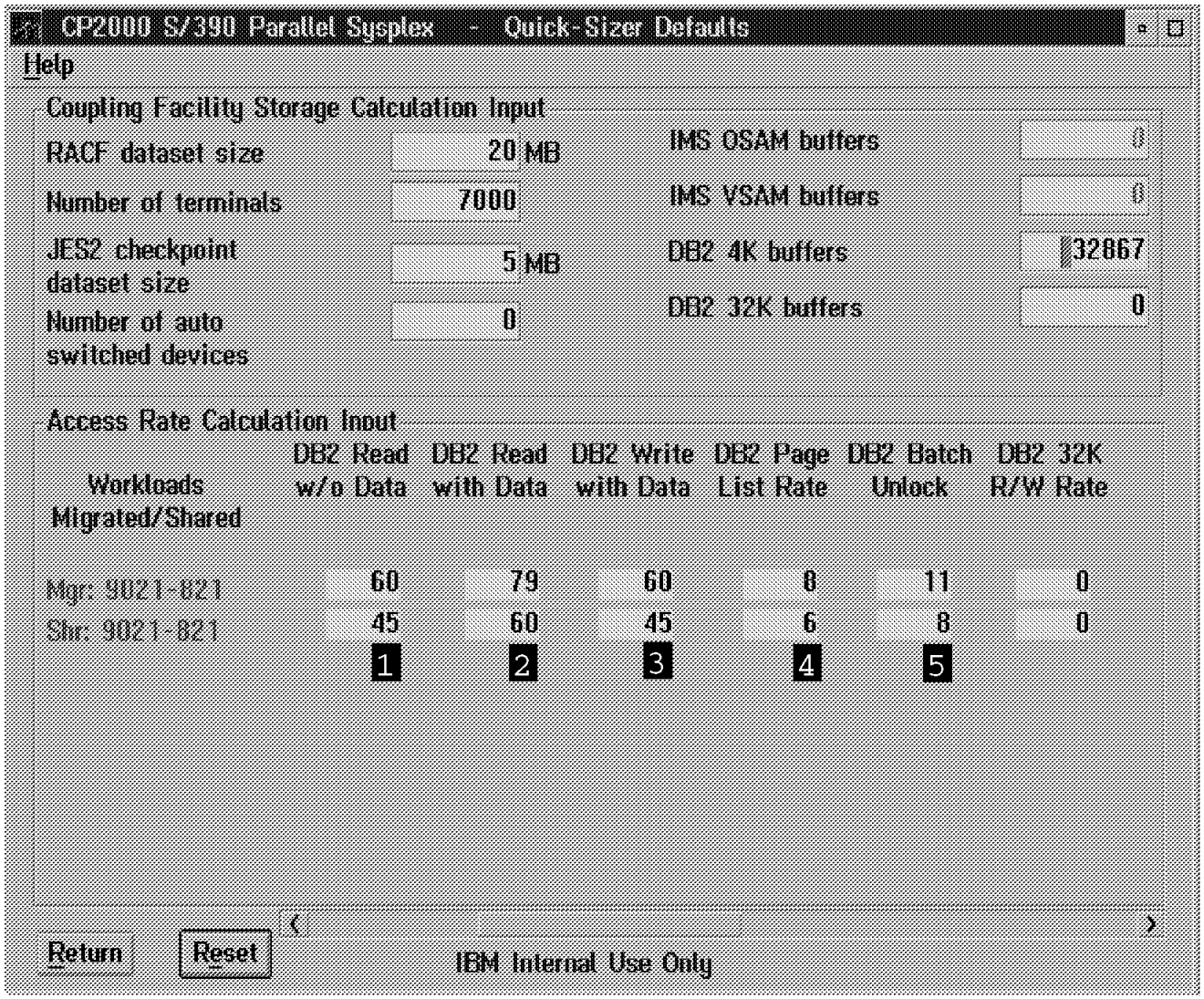


Figure 61. Quick-Sizer Defaults for DB2

This section explains which DP2PM fields you need for these calculations.

For each Quick-Sizer input field in Figure 61, you are presented with the corresponding Quick-Sizer assumption on how to use the DB2PM data.

For sample DB2PM reports, refer to Figure 71 on page 208 and Figure 72 on page 209. For examples on how to extract relevant numbers from DB2PM reports, do the necessary computation and then feed the results into Quick-Sizer (refer to 7.1, “Case Study: DB2 Migration to a Data Sharing Environment” on page 197).

In the following description, the left side of each equation represents the name of one of the input fields, and the right side shows how to compute it, with DB2PM data filed names capitalized.

1 DB2 Read w/o Data = *SYNCHRONOUS READS*.

This rate represents the buffer pool read requests that result in DASD I/Os, as the buffer was not found in the CF.

This number is also used to calculate the access rate to the cache directories in the CF, because each physical read results in a registration of the buffer in the cache. Due to the "batch register list" function, the number of CF accesses for registration is much smaller than this number.

How the *SYNCHRONOUS READS* number is obtained from DB2PM reports is discussed in 7.1, "Case Study: DB2 Migration to a Data Sharing Environment" on page 197.

2 DB2 Read With Data =

$$(\textit{Cache Write With Data} \times \frac{n-1}{n} + \textit{Cache Write With Data} \times 0.8)$$

Note that *n* represents the number of DB2 systems in the data sharing group.

Cache Read with Data occurs either for buffer pool read miss requests that do not result in DASD I/O because the buffer was found in the CF, or for cast-out processing.

Notice that the term Cache Write With Data is calculated in **3** later.

Term 1 represents the number of assumed reads from the CF.

Term 2 represents the number of assumed asynchronous writes to DASD after reaching the DB2 castout threshold.

We assume that $\frac{(n-1)}{n}$ represents the chance that the data in the CF was written last by *another DB2,* and hence is no longer in the local buffer pool because it would have been invalidated.

We also assume that 0.8 allows for the fact that a GBP buffer can be updated in the CF before castout processing takes place. The factor 0.8 means that, on average, 20% of the data written to the CF is changed one or more times before being read again for castout.

3 DB2 Write With Data = *BUFFER UPDATES* × 0.5

BUFFER UPDATES represents the number of times data has changed. 0.5 is an experience-based number. It represents the situation when a changed page is updated again before being committed. It is, in other words, assumed to be updated almost twice on average, before a COMMIT. Upon COMMIT and R/W Interest, the page is written to the CF. Notice the assumption of *R/W interest for all updates,* thus assuming 100% data sharing for the DB2 plans reflected in this report.

How the *BUFFER UPDATES* number is obtained from DB2PM reports is discussed in 7.1, "Case Study: DB2 Migration to a Data Sharing Environment" on page 197.

4 DB2 Page List Rate = *PAGES READ ASYNCHR* / 24

This represents the rate at which new pages are being registered in the CF.

As an assumption, the number 24 represents the number of pages per asynchronous read, based on experience. In reality, the maximum of possible pages per asynchronous read is 32.

How the *PAGES READ ASYNCHR* number is obtained from DB2PM reports is discussed in 7.1, “Case Study: DB2 Migration to a Data Sharing Environment” on page 197.

5 DB2 Batch Unlock = (*LOCK requests* – *UNLOCK requests*) × 0.1 × *m*

This represents the rate at which pages are being batch unlocked. Quick-Sizer has already counted unbatched unlocks in its locking rate figure. See 7.1.2.1, “Current Locking Rate Calculation (DB2PM)” on page 204 for details.

The factor 0.1 represents 1 batch unlock request for every 10 locks.

m is a “migration factor” to DB2 V4 Index Type 2.

For a discussion on *m*, refer to 2.2.4, “DB2 Locking Rate Calculations” on page 67.

6.3.1.1 Quick-Sizer CF Storage Calculations

The number of DB2 4K (and 32K) buffers is entered in one of the default panels (see Figure 61 on page 186). Quick-Sizer estimates the GBP size required by knowledge of the read/write activity to the CF described in the optional input fields.

6.3.2 IMS/DB CF Access Rate Calculation

Access Rate Detail Input

You can enter more detailed information about your workload to Quick-Sizer for CF access rate calculations. Figure 62 on page 189 describes that window for IMS/DB. To get access to this window, you have to specify IMS/IMS in Optional Input. Then you can reach this window by selecting **Defaults.** This panel shows an IMS/DB workload.

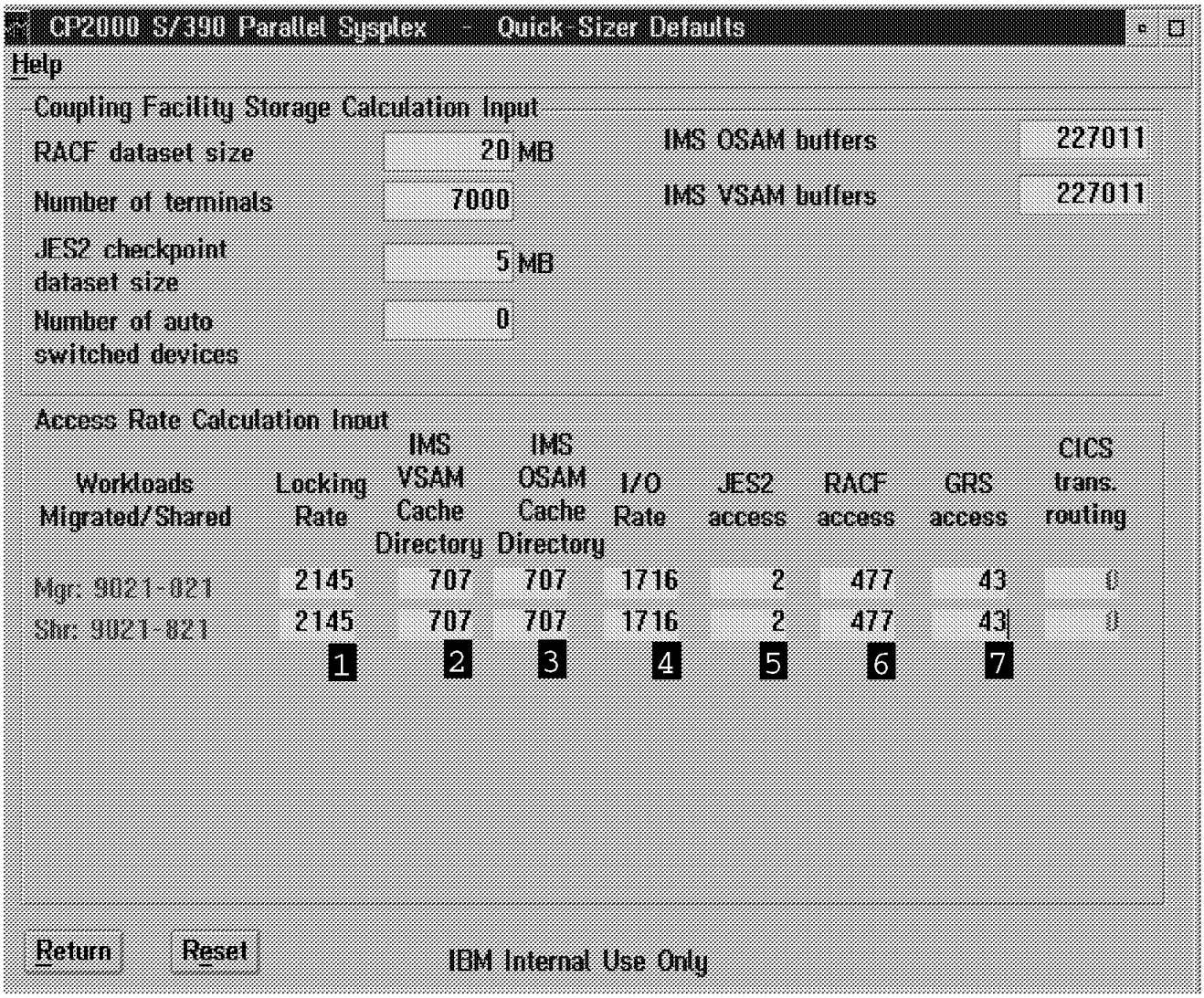


Figure 62. Quick-Sizer Defaults for IMS

To get information about the relative influence of the access rate components (**1** to **7**) on the total cost, refer to 2.1.5, “Cost Factors in Parallel Sysplex” on page 62.

The window in Figure 62 is divided into two parts:

- Coupling Facility Storage Calculation Input

 This input is used by Quick-Sizer to size the structures of the CF. Quick-Sizer adds storage for HSA, CFCC, dumpspace, and “white space” before showing the total amount of storage.

 “White space” is storage in a CF set aside for rebuilding structures from another CF in case of failure.

- Access Rate Calculation Input

 Quick-Sizer uses the values in this panel to estimate the utilization of the CEC and the CF. There are lines describing the workload that migrates (MGR) and others describing the workload that remains (SHR) per each current CEC, if any.

The following topic explains the different fields for IMS/DB Access Rate calculations. The window shown in Figure 62 defines detailed input that may be adjusted to arrive at a more precise projection.

1 Locking Rate

Number of accesses from IRLM that call XES locking to access the CF. You can get the appropriate IRLM accesses via the IRLM trace. How to calculate locking rates is discussed in 2.3.4, "IMS/DB Locking Rate Calculations" on page 78. Note that the rates often contribute the most to the Parallel Sysplex data-sharing cost.

2 IMS VSAM Cache Directory

This is the number of accesses per second from VSAM buffers to the CF. In a Parallel Sysplex, you can obtain the numbers from the RMF XCF Activity report.

In a non-sysplex environment, this information is not readily available; you may find VSAM read I/O rates reported in IMS tools such as OMEGAMON or IMSPARS.

Quick-Sizer uses a default value of 40% of the total I/O rate (**4**).

3 IMS OSAM Cache Directory

This is the number of accesses per second from OSAM buffers to the CF. In a Parallel Sysplex, you can obtain the numbers from the RMF XCF Activity report.

In a non-sysplex environment, this information is not readily available; you may find OSAM read I/O rates reported in IMS tools such as OMEGAMON or IMSPARS.

Quick-Sizer uses a default value of 40% of the total I/O rate (**4**).

4 I/O Rate

The database I/O rate is not always easy to obtain. You may be able to get it from RMF reports or appropriate monitors.

5 JES2 access

You can get the number of I/O operations accessing the JES2 checkpoint data set from the RMF DASD Activity Report and RMF XCF report. The RMF DASD Activity Report only shows this number if the checkpoint data set is the only data set on the volume.

6 RACF access

Number of accesses per second on a RACF data set. You can calculate these numbers from online monitors, QCBTRACE, or via RMF DASD reports.

7 GRS access

This input is obtained from the XCF activity report. Under the heading XCF USAGE BY MEMBER, find the member representing XCF and the GROUP defined for GRS. Under the column REQ OUT, locate the number for the GROUP defined for GRS to the member representing XCF. Divide this number by the length of the INTERVAL in seconds.

6.4 Other Parallel Sysplex Capacity Planning Issues

The following sections contain additional information on topics related to Parallel Sysplex capacity planning projections:

- CPU projections
- Storage projections
- I/O projections
- CF projections

Each section explains or provides pointers to other sections that show how CP2000 may assist you.

6.4.1 Workload and CPU Projections

Discussions on how to estimate the CPU cost for participating in Parallel Sysplex data sharing configurations are found in:

- 2.2.3, “Parallel Sysplex CPU Cost for DB2 Data Sharing” on page 65.
- 2.3.3, “Parallel Sysplex CPU Cost for IMS/DB Data Sharing” on page 77.

This section offers a short explanation of the capabilities of CP2000 regarding workload projections on different CECs.

You identify:

- The workloads to be considered for future use
- The projected growth rates for the workloads
- The processors to be considered for future use

CP2000 gives you several opportunities to forecast the future CPU power you need. You can let your system grow with different percentages for separate workloads and with variable growth percentages over different periods in time. Disappearance of workloads over time, or adding new workloads at any time you want, is also a possibility. You can do this for three different growth scenarios.

In each scenario, there is a possibility to evaluate three different processors which you can introduce at many points in time. You can also adjust the saturation design point to the value that best fits your environment to make the projections more accurate. Remember that capacity planning is an iterative process. It is easy to change parameters, expectations and time periods to suit your needs. You also have options to introduce new CECs at points in time of your choice.

CP2000 gives you great flexibility to make a capacity plan. The output contains a variety of different graphs that give you different views of your processor needs in the coming years.

Try It Out!

You are encouraged to try CP2000. For your convenience, there is a CP2000 case study that will take you through most of the steps listed above. *You do not have to provide your own data*, since it is an integral part of the case study. Refer to Chapter 7, “Parallel Sysplex Capacity Planning Case Studies” on page 195.

6.4.1.1 Engine Size Issues

In the Parallel Sysplex, when migrating to a CEC with a higher processing power, the central processors (CPs) in the new CEC were usually faster than the old ones.

When you migrate to a Parallel Sysplex and 9672 CECs, either with *lower or higher* combined processing power, the central processor (CP) of the 9672s may not be as fast as the CP in the old CEC. If certain applications are only able to use a single CP, you have to be aware that a CP service time increase may increase your response time. Also, your single-threaded application may not fit at all if it requires more processing power than a single CP can provide.

CP2000 can model CP service time.

For a discussion of the Analyze processor speed option, refer to 7.2, “Case Study: Analyze Processor Speed” on page 215.

For more information, IBM representatives may refer to *MVS Performance Capacity Planning Considerations for 9672-Rxx Processors*, WSC Flash 9505.

A suite of tools and services is available to IBM representatives for detailed studies. For more information on these and other tools and services, refer to A.6, “Enterprise Server Capacity Planning Tools Catalogue” on page 271.

6.4.2 Storage Projections

Discussions on how to size buffer pools in Parallel Sysplex data sharing configurations are found in:

- 2.2.5, “DB2 Buffer Pool Sizing in Parallel Sysplex” on page 70
- 2.3.5, “IMS/DB Buffer Pools Sizing in Parallel Sysplex” on page 80

For a discussion on whether to use expanded storage or central storage, refer to 1.5, “Capacity Planning and Storage” on page 35.

Depending on your currently installed amount of storage and whether it is in good balance with the processor resources consumed today, you can extrapolate your storage requirement for the future. If your storage is not in balance, CP2000 can provide the opportunity to optimize it with a factor called the *storage calibration factor*. Underlying this calibration factor are rules that advise you how much storage you should need today. This happens either if you have excess storage or not enough storage. When you finish the calibration (if necessary), you can make your projections for the processor storage needed. CP2000 gives you an estimation of the processor storage you need for the next two years and does not make a distinction between central and expanded storage. The storage extrapolation is based on simple algorithms. The formula used by CP2000 gave the relationship between the M-value and the amount of storage (central and expanded) needed:

$MB = K1 + 0.01 \times (M\text{-}value \times B)^d$ where:

- MB is the amount of storage.
- K1 is a value that varies with the level of the operating system. K1 is the minimum working set size for a particular operating system.
- M-value is the CP2000 M-value.
- B (busy) is the average CPU utilization.

- d is a constant unique to each installation.

The idea behind this formula is to make a “curvefit” that resembles your installation.

If we look at the range of observed storage sizes over the M-value used (single OS/390 system), the curve looks like Figure 63. The range is based on statistics collected by CP2000. The upper curve (d=1.58) illustrates S/390 installations with a relatively large amount of installed storage, and the lower curve shows the same for a relatively small amount of installed storage.

Installations in the upper part of the diagram represent the case where applications tend to use large in-storage buffers and other data-in-memory techniques. Installations that are over-configured also belong to this part of the diagram.

Installations in the lower part of the diagram represent recent migrations to OS/390, or installations that are on the borderline of being storage constrained.

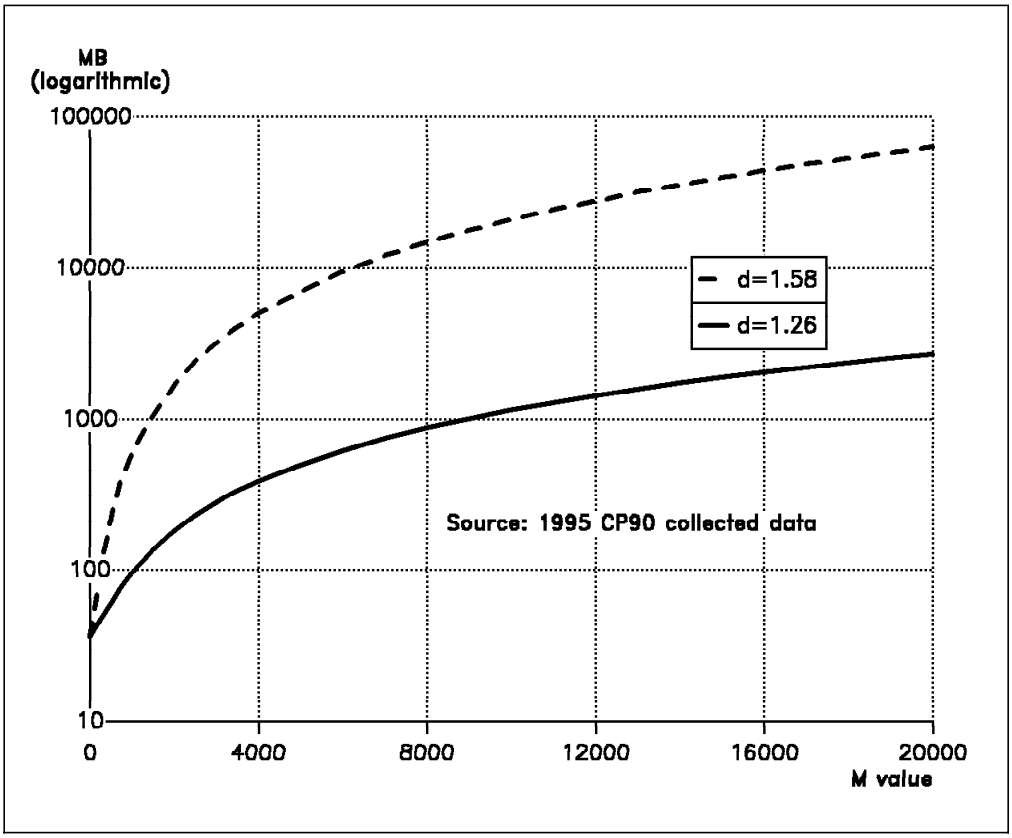


Figure 63. The Range of Storage Sizes over the M-Value

For more information about storage calculations, refer to *Balanced Systems and Capacity Planning*, GG22-9299, and “IBM Services” described in section “MVSSTOR” in A.6, “Enterprise Server Capacity Planning Tools Catalogue” on page 271.

CPAIP provided information about the value of d in the so-called metric report “Summary Metrics.” In CPAIP, the constant d was called the “Main Storage Calibration Constant.”

6.4.3 I/O Projections

Discussions on how to estimate the I/O effect due to buffer invalidation in Parallel Sysplex data sharing configurations are found in:

- 2.2.2, “No Expected I/O Growth for DB2” on page 64
- 2.3.2, “Expected I/O Growth for IMS/DB” on page 76

Chapter 7. Parallel Sysplex Capacity Planning Case Studies

At a Glance

This chapter proceeds step-by-step through several Quick-Sizer actions required to create a capacity plan. It is written based on several Quick-Sizer case studies, and it helps you understand how to use Quick-Sizer to create your own capacity plan.

Recommended Reading:

- *CP2000 S/390 Parallel Sysplex Quick-Sizer User's Guide*

© Copyright IBM Corp. 1998

Three case studies are presented in this chapter:

7.1, "Case Study: DB2 Migration to a Data Sharing Environment" on page 197

This is a DB2 case study using the Quick-Sizer. The goal of this case study is to *model DB2 data sharing.* Included is a discussion on what input options Quick-Sizer provides and how this information is obtained from reports, such as DB2PM.

To go through this case study you need *less than an hour* if you already have the OS/2 version of Quick-Sizer installed on your PC or server. To install Quick-Sizer, you need an additional 30 minutes.

7.2, "Case Study: Analyze Processor Speed" on page 215

This case study is an analysis of the transaction response times when workloads are moved to another CEC. In this case study, you will be using the CP2000 PC-based tool, so you need to have it installed on your workstation.

7.3, "Case Study: CF Modelling" on page 226

This case study uses the CF model introduced in Chapter 5, "A CF Processing Model" on page 143 to discuss the effects of changing the CF versus the link configuration in a concrete environment. The spreadsheet can be obtained from IBM internally and you can conclude the case study in less than half an hour.

If you do not have the spreadsheet, you can still use the case study as a comprehensive example and discussion for the CF model.

7.1 Case Study: DB2 Migration to a Data Sharing Environment

This case study covers a DB2 (IMS or CICS transaction manager) workload migration to a Parallel Sysplex. We arbitrarily refer to a CICS/DB2 workload in this exercise. Our single system runs on a single CEC and has three types of workloads:

- The first part of the CICS/DB2 workload will be migrated to 9672 systems and will share the DB2 database.
- The second part of the CICS/DB2 workload will remain on the current system and share the DB2 database.
- The third workload element remains on the current system. It is not related to the DB2 workloads previously described.

We use the Quick-Sizer for this study. If you are not familiar with this tool, refer to *CP2000 S/390 Parallel Sysplex Quick-Sizer User's Guide* for further information.

The single system today (9021-821) has, at peak period, a workload activity as follows:

- 60% CPU usage serving the CICS/DB2 workload with the following characteristics:
 - Part of the workload (40% CPU) to be moved off to 9672s and share the DB2 database.
 - Part of the workload (20% CPU) to remain and share the DB2 database.
 - The total workload sustains a load of 72 transactions per second.
 - The total workload creates 600 DASD I/Os per second.
- 30% CPU utilization is used today on the 9021-821 to serve other workloads. These remain on the 9021-821 and do *not* share the DB2 database.
- The total CPU load today on the 9021-821 is 90% (30% + 60%). We assume the techniques discussed in 1.3.2.1, “Capture Ratio” on page 15 have been used to assign all used CPU to applications.

Figure 60 on page 181 shows the current and target environments for a similar migration. (Note that the numbers in the figure *do not match* those of our case study, this figure is shown for comparison purposes only).

In addition to the installation details previouslyu listed, we also require a set of DB2PM reports to provide additional information about the DB2 subsystem usage, including locking activity. Note that it is possible to use the Quick-Sizer default values for DB2-related CF access rates. DB2PM reports are then not needed. If Quick-Sizer default values are used, the results should be used with caution, since your actual CF request rates may differ from the Quick-Sizer default values.

We also assume that we have a CICS transaction rate of 29 per second, and an I/O rate to our database of 600 I/O per second. These figures are not essential, but thay do enable us to have more confidence in the model.

7.1.1 Quick-Sizer Default Input and Output

Define the Current Configuration

Quick-Sizer is an IBM internal tool. IBM customers may obtain the output from this tool through their IBM representative.

After having installed and activated the latest version of Quick-Sizer you will see a window similar to Figure 64 page=no.. You will see the CP2000 Quick-Sizer welcome window. Click **OK**. We now fill in the fields to achieve the results shown in Figure 64.

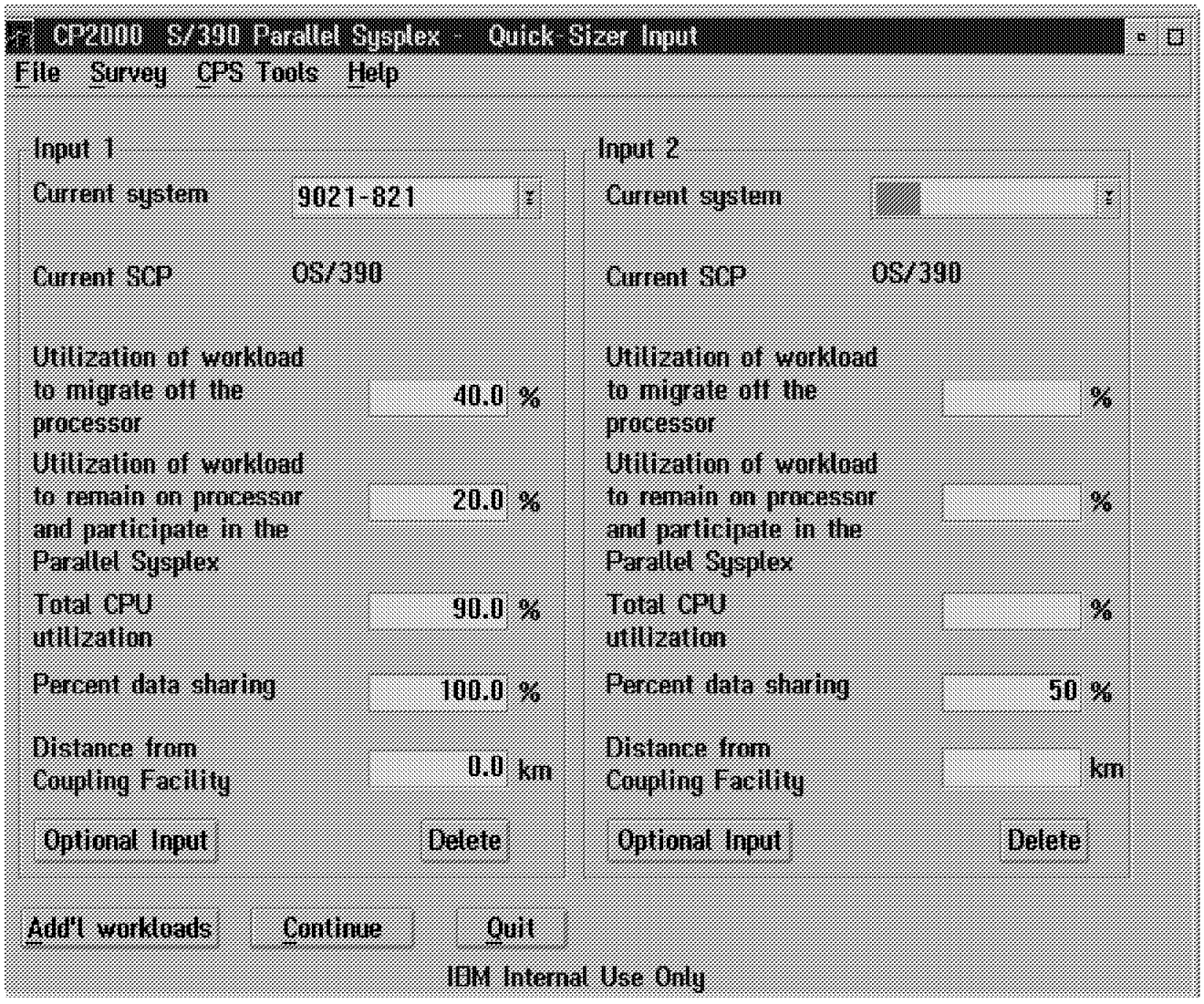


Figure 64. Quick-Sizer Input Window

First, we have to describe the current configuration with respect to its workload elements. For a detailed discussion about this, refer to:

- 6.2.1, “Defining Your Environment to Quick-Sizer” on page 165
- 6.2.2, “Quick-Sizer Parallel Sysplex Migration” on page 180

The actions are as follows:

1. For Current system, select **9021** first and then select **9021-821**.

 This field describes the current CEC running the applications to be moved to a Parallel Sysplex environment. A drop-down list contains all CECs supported.

2. Current SCP is fixed as OS/390.

 This field was selectable in earlier versions of the tool.

3. For Utilization of workload to migrate off the processor, enter 40.

 This field represents the current CEC utilization (expressed as CPU busy %) for the CICS/DB2 workload elements to be moved off the current CEC to other CECs within the Parallel Sysplex. These workload elements are eligible for Parallel Sysplex data sharing. Be sure to account for uncaptured CPU time, as well as OS/390, VTAM, RACF, and so forth, being used by this workload.

4. For Utilization of workload to remain on processor and participate in the Parallel Sysplex, enter 20.

 Enter the current CEC utilization (expressed as CPU busy %) represented by the CICS/DB2 workload elements that will remain on the current CEC and are eligible for sharing the DB2 database in the Parallel Sysplex. This utilization only represents the workload components that are eligible to participate in a Parallel Sysplex implementation. Be sure to account for uncaptured CPU time, as well as OS/390, VTAM, RACF, and so forth, being used by this workload.

5. For Total CPU utilization, enter 90.

 This field represents the *total utilization* of the current CEC (expressed as CPU busy %).

6. For Percent data sharing, enter 100.

 All the transactions in the CICS/DB2 workload are eligible for sharing the DB2 database. There are no other databases or application data sets for these transactions. We select a data sharing percentage of 100.

 We use our own values from supplied DB2PM reports, instead of the default values provided by Quick-Sizer. This is one reason for specifying data sharing as 100%. If you specify a value lower than 100% for data sharing and replace the default values with your own values, Quick-Sizer reduces your own values according to the data sharing percentage.

 Only use Quick-Sizer′s less than 100% data sharing option when you use the Quick-Sizer′s default values (for locking and so on). For a description on how Quick-Sizer uses the “Percent data sharing” option, refer to 6.2.2, “Quick-Sizer Parallel Sysplex Migration” on page 180.

7. For Distance from Coupling Facility, enter 0.

 The CF will be located very close to today′s system.

8. Click on **Optional Input** to go to Figure 65 on page 200.

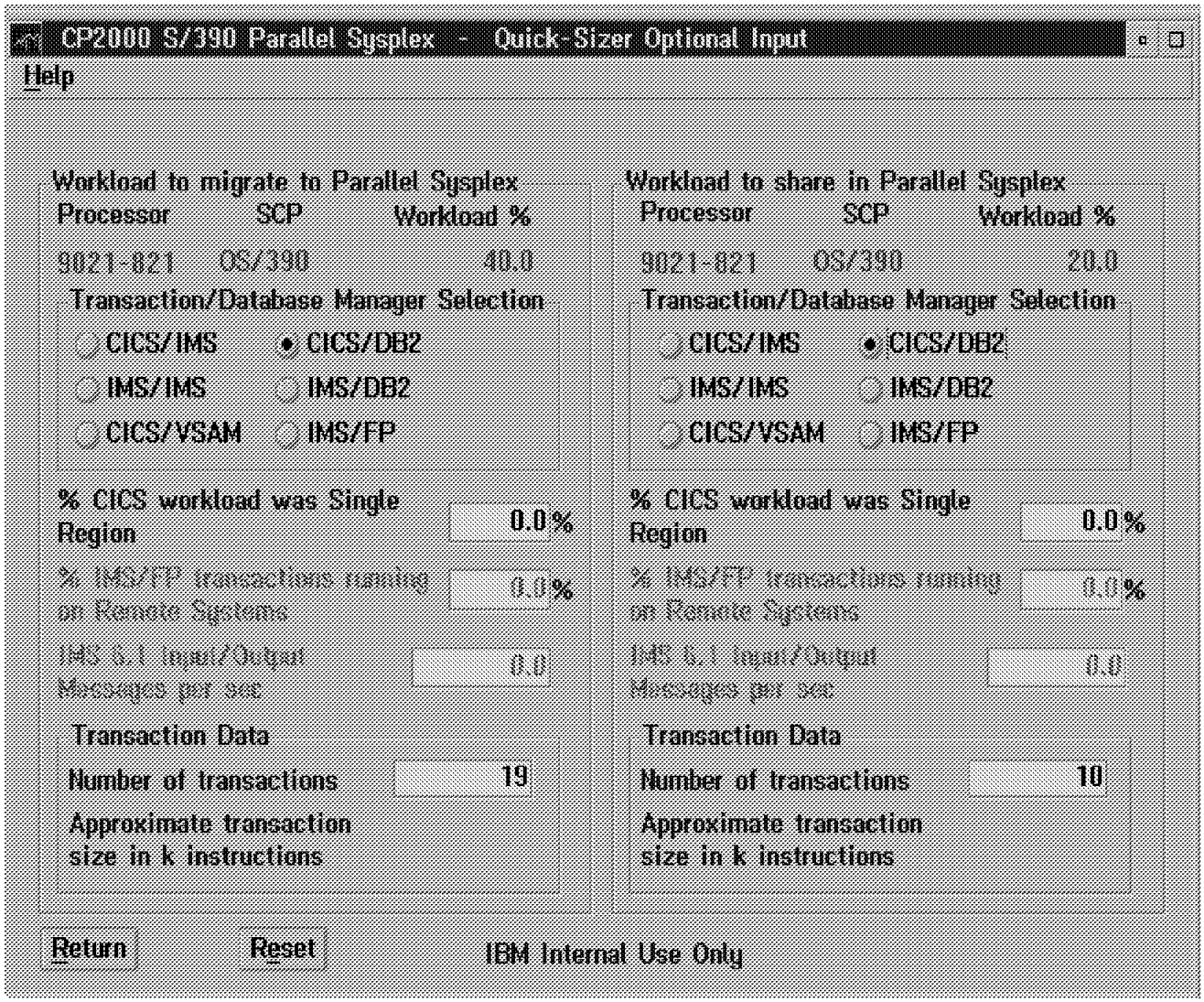


Figure 65. Quick-Sizer Optional Input Window 2

Your next set of actions are:

1. Click on **CICS/DB2** in the Workload to migrate to Parallel Sysplex section of the window.
2. Click on **CICS/DB2** in the Workload to share in Parallel Sysplex section of the window.
3. Check the Number of transactions field.

 CP2000 Quick-Sizer assumes two and a half million instructions per transaction for CICS/DB2. The number of transactions is calculated accordingly. In our case, the numbers “came out right” since we knew that the transaction rate for our workload was 29 transactions per second. This is exactly the sum of 19 and 10 that shows in Figure 65. If you have a different transaction rate, change these numbers.

 Quick-Sizer uses the transaction rate numbers for the response time calculations only. When you change the transaction rate number, Quick-Sizer displays the path length that your transaction rate implies. The numbers are also useful as a reality check.

4. Click on **Return**.

This returns you to the Input window shown in Figure 64 on page 198.

1. Click **Continue**.

Now you see the Output window shown in Figure 66 on page 202.

Before executing the evaluation, you may modify the following fields:

- 9672 Saturation design point

 Enter the upper limit for average CEC utilization here; a default of 70% is provided, which may be modified. This value should be chosen so that peak period utilizations can be sustained without any significant impact on response time.

- Coupling Facility target utilization

 Enter the upper limit for average CEC utilization here; a default of 50% is provided, which may be modified. This value should be chosen so that peak period utilizations can be sustained without any significant impact on response time. It may be a good idea to specify a smaller value to avoid queuing for the CF. Queuing for the CF increases the service time and thereby increases the CEC CPU cost in the CICS/DB2 system. (See also the description of Redundancy which follows.)

- Distance of 9672 from the Coupling Facility

 This field may be used if your 9672 is distant from the CF.

- Other CF

 Specify this field if you wish to choose a specific CF model. Remember that the Quick-Sizer uses either a 9672 model to denote a 9674 model (for example, 9672-R22 means a 2-way 9674-C02), or a 2003 Multiprise model to denote the 1-5 way 9674-C04 (for example, 9674-156 really means a 5-way 9674-C04).

- Systems already part of sysplex

 Clicking on this field reduces the cost added on for multisystem management.

- Redundancy

 If CF redundancy is required for availability reasons, check the box. Note that if you choose redundancy, Quick-Sizer assumes the target utilization should apply in the event of the loss of one CF.

- Parallel channels

 Check this box if you plan to have parallel channels in your Parallel Sysplex configuration.

- IMS 6.1

 Check the **IMS 6.1** button off. (To turn it on if you had made a mistake, you would need to go back to step 8 on page 199 and then click on an IMS option, as shown in Figure 65 on page 200, instead of on **CICS/DB2**.)

- IMS SMQ

 As shown in Figure 66 on page 202, the **IMS SMQ** button is greyed out now. To select it, you would have to reselect **IMS 6.1**.

- OSAM Caching

 As shown in Figure 66, the **OSAM Caching** button is greyed out now. This is because OSAM is not relevant to this configuration.

- GRS Star

 Select the button for **GRS** star.

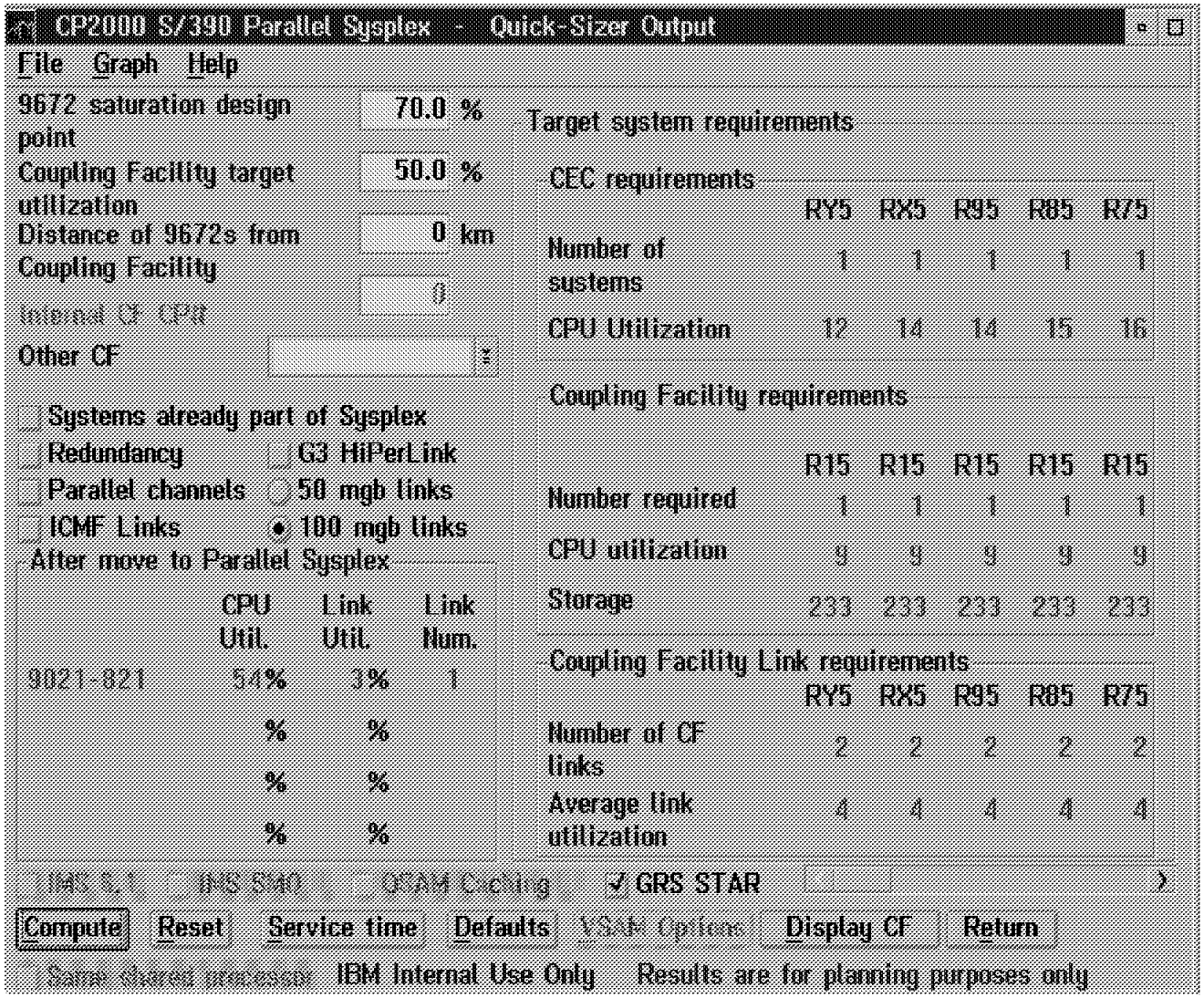


Figure 66. Quick-Sizer Output Window

As shown in Figure 66, do the following:

1. Click the **100 mgb links** button.
2. Click on **Compute**. Once you have provided the characteristics of the projected environment, the results are calculated. The window in Figure 66 may already contain some values in the output fields. The numbers in your window may be different from the ones shown here.
3. Click on **Display CF** for details of storage and CF usage estimates.

7.1.2 Quick-Sizer Detailed Input and Output

Next, we go a little further and obtain more accurate results, using some additional information we have collected as shown in Figure 66 on page 202. Select the **Quick-Sizer Defaults** panel to continue. The numbers you see in this window may be different from the ones shown because the numbers you get are calculated based on Quick-Sizer default values. To verify and display these values, click on **Defaults**. Now you see the Defaults window in Figure 67.

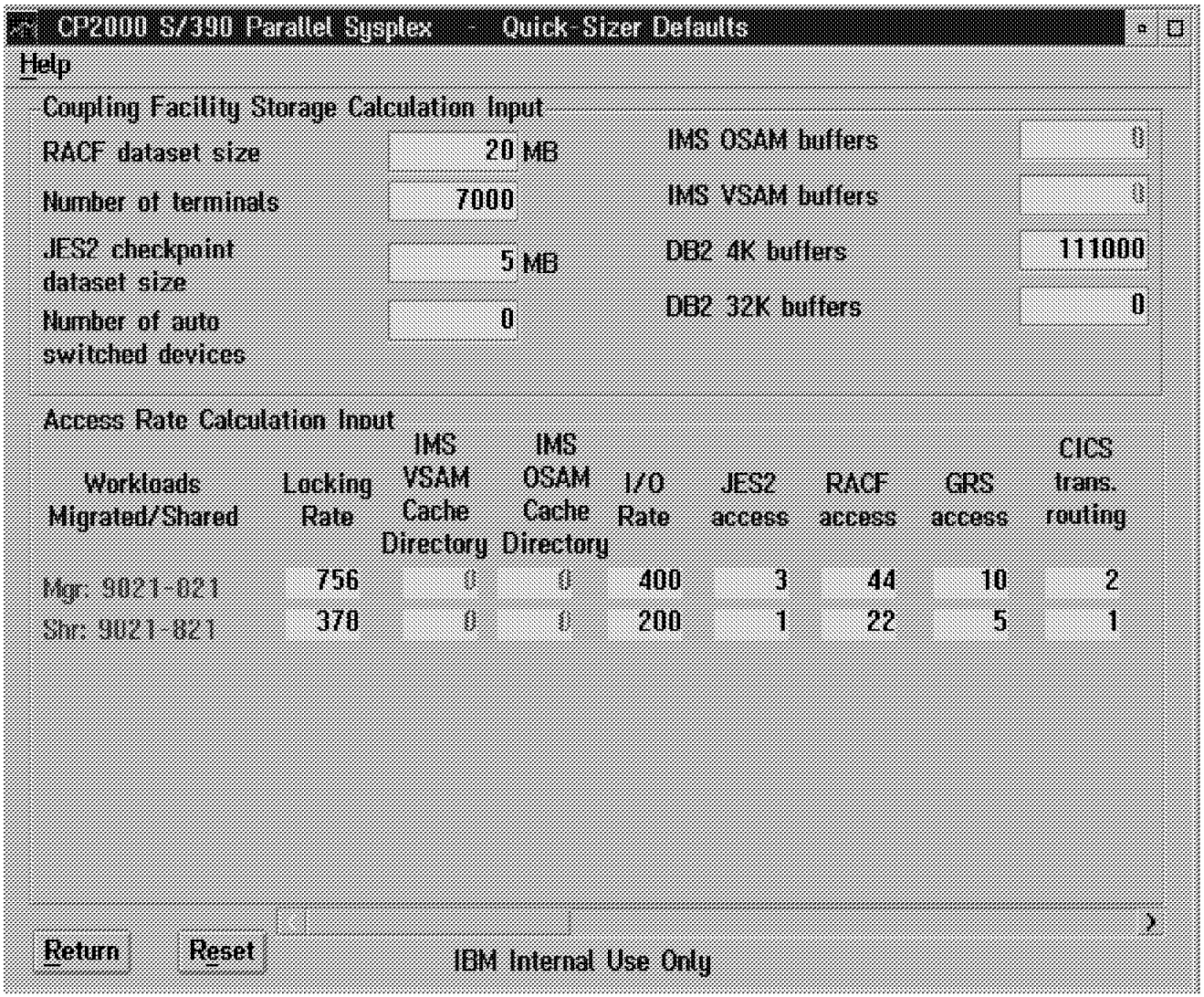


Figure 67. Quick-Sizer Defaults Window (Scrolled Left). Some of the window values have already been updated to the installation requirements.

You are now requested to change these values to represent your situation. Figure 67 shows the already updated values; we will explain how we obtained these numbers. Note the default Quick-Sizer CF usage of both JES2 checkpoint and RACF. In this example we leave these values as they are.

7.1.2.1 Current Locking Rate Calculation (DB2PM)

What Quick-Sizer expects for locking rate

The number to enter in the **Locking Rate** field is the LOCK REQUEST rate plus the UNLOCK request rate plus the CHANGE request rate from DB2PM. The capitalization indicates the names of fields on DB2PM reports.

The remainder of this section explains where to find these numbers and how to adjust them to allow for differences in starting and ending DB2 levels and index options. The unlocks unaccounted for in this number are handled by batch unlocking and a number is entered in a separate field (on the next panel).

The locking activity creates a significant part of the request activity to the CF.

If DB2PM is not used, you may use other equivalent monitors.

For a peak period of two hours, let us assume DB2 traces were run for the CICS/DB2 workload and the corresponding DB2PM reports were created. To get the required data, activate the following DB2 traces and collect the information from SMF records type 100 and 101:

```
TRACE(ACCTG) CLASS(1,2,3)
TRACE(STATS) CLASS(1,2,3)
```

Statistics reports with classes 1 and 3 may be sufficient. Consult your DB2 specialist for the details.

For more information about the DB2 statistics and accounting SMF records, refer to *DB2 for OS/390 V5 Administration Guide*, SC26-8957. To keep things reasonably simple, you will only be shown a few excerpts from the DB2PM reports. We migrate the entire DB2 workload to the Parallel Sysplex. The numbers in the DB2PM reports are *totals*, and thus can be used directly after a simple conversion to *rates*.

Figure 68 and Figure 69 on page 205 show two excerpts from DB2PM reports with locking numbers. You only need one of these reports, but we show both here. It does not matter which of the reports you use.

```
ACCOUNTING REPORT

LOCKING          AVERAGE     TOTAL
--------------  --------  --------
TIMEOUTS             ---       ---
DEADLOCKS            ---       ---
ESCAL.(SHARED)       ---       ---
ESCAL.(EXCLUS)       ---       ---
MAX LOCKS HELD       ---       ---
LOCK REQUEST         ---   9702524 1
UNLOCK REQUEST       ---   6141603 2
QUERY REQUEST        ---       ---
CHANGE REQUEST       ---    482418 3
OTHER REQUEST        ---       ---
LOCK SUSPENS.        ---       ---
LATCH SUSPENS.       ---       ---
OTHER SUSPENS.       ---       ---
TOTAL SUSPENS.       ---       ---
```

Figure 68. DB2PM Accounting Report (Lock Information)

```
STATISTICS REPORT

LOCKING ACTIVITY             QUANTITY
-------------------------- ----------
SUSPENSIONS (ALL)                 ---
SUSPENSIONS (LOCK ONLY)           ---
SUSPENSIONS (LATCH ONLY)          ---
SUSPENSIONS (OTHER)               ---

TIMEOUTS                          ---
DEADLOCKS                         ---

LOCK REQUESTS                9702524 1
UNLOCK REQUESTS              6141603 2
QUERY REQUESTS                    ---
CHANGE REQUESTS               482418 3
OTHER REQUESTS                    ---
```

Figure 69. DB2PM Statistics Report (Lock Information)

The numbers we use from the report in Figure 68 on page 204 (which are similar to the corresponding numbers in Figure 69.) have to be divided by the length of the period. This gives us the rate expressed as requests per second. The time period in our case is two hours (7200 seconds).

Extracting the relevant request numbers from the DB2PM reports and dividing by 7200 yields:

$$\textbf{1}\ \textit{LOCK REQUEST rate} = \frac{9702524\ \textit{requests}}{7200\ \textit{seconds}} \approx 1348\ \frac{\textit{requests}}{\textit{second}}$$

$$\textbf{2}\ \textit{UNLOCK REQUEST rate} = \frac{6141603\ \textit{requests}}{7200\ \textit{seconds}} \approx 853\ \frac{\textit{requests}}{\textit{second}}$$

$$\textbf{3}\ \textit{CHANGE REQUEST rate} = \frac{482418\ \textit{requests}}{7200\ \textit{seconds}} \approx 67\ \frac{\textit{requests}}{\textit{second}}$$

The sum of these three rates is now modified according to the “rules” outlined in 2.2.4, “DB2 Locking Rate Calculations” on page 67. and 6.3.1, “DB2 CF Access Rate Calculation” on page 184.

7.1.2.2 Future Locking Rate Calculation (DB2PM)

In our case, suppose we have DB2 V3 with Index Type 1 on the current system. In the Parallel Sysplex, we use DB2 V5 with Index Type 2. This is the current recommendation and assumption for all Quick-Sizer DB2 data sharing modelling. We therefore select the migration factor *m* based on the discussion in section 2.2.4, “DB2 Locking Rate Calculations” on page 67, so m=0.5. In other words, we expect the locking rate to be halved when we move from DB2 V3 to DB2 V5 with Type 2 indexes.

We multiply the DB2 V3 locking rate by *m* to get the V5 locking rate:

$$V5\ \textit{Locking rate} = m \times V3\ \textit{Locking rate} \approx 0.5 \times (1348 + 853 + 67) \approx 1134\ \frac{\textit{lock requests}}{\textit{second}}$$

This is the value Quick-Sizer expects for the locking rate. It has to be apportioned into two parts according to the ratio of our workload split between today′s system and the 9672 systems, which is one to two (20% to 40%).

The split approximates:

- 378 for the 9021-821
- 756 for the 9672s

Let us again return to CP2000 Quick-Sizer.

In the window in Figure 67 on page 203, your actions are:

1. For DB2 4K buffers, enter 111000.

 This field was not discussed previously. In the Parallel Sysplex we use a 434MB (111000 4K buffers) buffer pool in each system. The access pattern is the same in each system, so the size remains unchanged. This number is used by Quick-Sizer for the calculation of CF storage (for the GBP).

 How to size DB2 buffer pools in a Parallel Sysplex is discussed in 2.2.5, “DB2 Buffer Pool Sizing in Parallel Sysplex” on page 70.

2. For Locking Rate for Mgr: 9021-821 (migrated off 9021-821 to 9672), enter 756.

 The number 756 was derived from the previous calculation.

3. For Locking Rate for Shr: 9021-821 (sharing in 9021-821 with 9672), enter 378.

4. For I/O Rate for Mgr: 9021-821 (migrated off 9021-821 to 9672), enter 400.

 The split of I/O is based on the split of the workload. As the workload split is one to two (20% to 40%), the split of I/Os is assumed to be the same. In fact, Quick-Sizer has prefilled the field with a number very close to 400 already. However, as mentioned earlier, Quick-Sizer is not very sensitive to this field.

5. For I/O Rate for Shr: 9021-821 (sharing in 9021-821 with 9672), enter 200.

6. Leave the other fields as they are.

7. Click on the horizontal scrollbar at the bottom right part of the panel to get more DB2 input fields like those shown in Figure 70 on page 207.

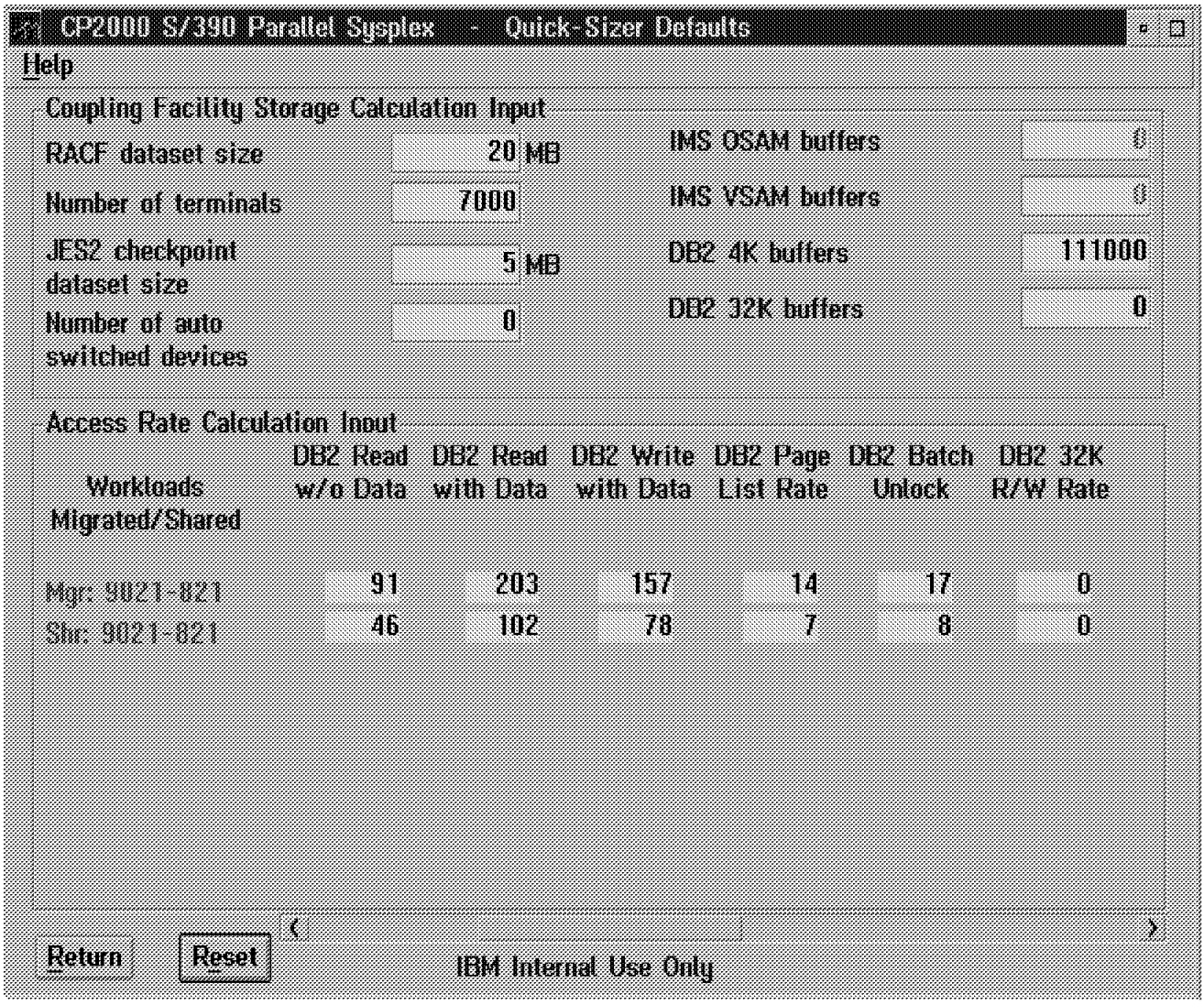


Figure 70. Quick-Sizer Defaults Window (Scrolled Right)

7.1.2.3 Further Statistics Required from DB2PM

Because the fields in Figure 70 have already been updated with the installation data, the numbers in the lower part of your window may be different from the ones shown here.

In this case, study the values that were extracted from the DB2PM reports in Figure 71 on page 208.

Figure 71 on page 208 and Figure 72 on page 209 show two DB2PM reports with buffer pool statistics numbers. You only need one of these reports, although we show both here. It does not matter which one you choose to use.

```
ACCOUNTING REPORT

TOT4K                AVERAGE     TOTAL
-------------------  --------  --------
EXPANSIONS               ---       ---
GETPAGES                 ---       ---
BUFFER UPDATES           ---   3374462 1
SYNCHRONOUS WRITE        ---       ---
SYNCHRONOUS READS        ---    987442 2
SEQUENTIAL PREFETCH      ---       ---
LIST PREFETCH            ---       ---
DYNAMIC PREFETCH         ---       ---
PAGES READ ASYNCHR.      ---   3653916 3
HPOOL WRITES             ---       ---
HPOOL WRITES-FAILED      ---       ---
PAGES READ-HPOOL         ---       ---
HPOOL READS              ---       ---
HPOOL READS FAILED       ---       ---
```

Figure 71. DB2PM Accounting Report (Buffer Pool Information)

```
STATISTICS REPORT


  TOT4K  READ OPERATIONS       QUANTITY
  ------------------------- ----------
  GETPAGE REQUEST                 ---
  GETPAGE REQUEST-SEQUENTIAL      ---
  GETPAGE REQUEST-RANDOM          ---

  SYNCHRONOUS READS            987442  2
  SYNCHRON. READS-SEQUENTIAL      ---
  SYNCHRON. READS-RANDOM          ---

  GETPAGE PER SYN.READ-RANDOM     ---

  SEQUENTIAL PREFETCH REQUEST     ---
  SEQUENTIAL PREFETCH READS       ---
  PAGES READ VIA SEQ.PREFETCH  2521876  4
  S.PRF.PAGES READ/S.PRF.READ     ---

  LIST PREFETCH REQUESTS          ---
  LIST PREFETCH READS             ---
  PAGES READ VIA LIST PREFTCH   36734  5
  L.PRF.PAGES READ/L.PRF.READ     ---

  DYNAMIC PREFETCH REQUESTED      ---
  DYNAMIC PREFETCH READS          ---
  PAGES READ VIA DYN.PREFETCH  1095306  6
  D.PRF.PAGES READ/D.PRF.READ     ---



  TOT4K  WRITE OPERATIONS      QUANTITY
  ------------------------- ----------
  BUFFER UPDATES               3374462  1
  PAGES WRITTEN                   ---
  BUFF.UPDATES/PAGES WRITTEN      ---

  SYNCHRONOUS WRITES              ---
  ASYNCHRONOUS WRITES             ---

  PAGES WRITTEN PER WRITE I/O     ---

  HORIZ.DEF.WRITE THRESHOLD       ---
  VERTI.DEF.WRITE THRESHOLD       ---
  DM CRITICAL THRESHOLD           ---
  WRITE ENGINE NOT AVAILABLE      ---

  SYNC.HPOOL WRITE                ---
  ASYNC.HPOOL WRITE               ---
  HPOOL WRITE FAILED              ---
  ASYN.DA.MOVER HPOOL WRITE-S     ---
  ASYN.DA.MOVER HPOOL WRITE-F     ---

  PAGE-INS REQUIRED FOR WRITE     ---
```

Figure 72. DB2PM Statistics Report (Buffer Pool Information)

The numbers we use from the report in Figure 71 on page 208 and the report in Figure 72 have to be divided by the length of the period. This yields the rate per second. The period length is two hours (7200 seconds).

Note: Data is from the DB2PM Accounting Report unless otherwise noted.

Note: The PAGES READ ASYNCH field in the DB2PM Accounting Report is the sum of the three fields in the DB2PM Statistics Report:

4 PAGES READ SEQ VIA SEQ. PREFETCH
5 PAGES READ VIA LIST PREFETCH
6 PAGES READ VIA DYN. PREFETCH

Extracting the relevant request numbers from the DB2PM reports and dividing by 7200 yields:

1 $BUFFER\ UPDATES\ rate = \frac{3374462\ requests}{7200\ seconds} \approx 469\ \frac{requests}{second}$

2 $SYNCHRONOUS\ READS\ rate = \frac{987442\ requests}{7200\ seconds} \approx 137\ \frac{requests}{second}$

3 $PAGES\ READ\ ASYNCHR\ rate = \frac{3653916\ requests}{7200\ seconds} \approx 507\ \frac{requests}{second}$

Note: This data is from the DB2PM Statistics Report.

4

$PAGES\ READ\ VIA\ SEQ.\ PREFETCH\ rate = \frac{2521876\ requests}{7200\ seconds} \approx 350\ \frac{requests}{second}$

Note: This data is only in the DB2PM Statistics Report.

5 $PAGES\ READ\ VIA\ LIST\ PREFETCH\ rate = \frac{36734\ requests}{7200\ seconds} \approx 5\ \frac{requests}{second}$

Note: This data is only in the DB2PM Statistics Report.

6

$PAGES\ READ\ VIA\ DYN.\ PREFETCH\ rate = \frac{1095306\ requests}{7200\ seconds} \approx 152\ \frac{requests}{second}$

Now that we have the different request rates computed, there are only a few more calculations left. You need to calculate the values to put into Figure 70 on page 207.

7.1.2.4 Calculations for Other CF Access Rates

Total CF Access Rate and Split

The numbers from DB2PM enable us to calculate the *total* CF access rates from the Parallel Sysplex systems. We have to prorate this in proportion to our workload split between the current system and the 9672 systems.

$\text{DB2 cache Read without Data} = SYNCHRONOUS\ READS\ rate \approx 137\ \frac{requests}{second}$

$\text{DB2 cache Read with Data} = (235 \times \frac{(2-1)}{2}) + (235 \times 0.8) \approx 305\ \frac{requests}{second}$

The number 235 comes from the next calculation. We assume two OS/390 systems in the Parallel Sysplex, each having one DB2 subsystem.

For a description of this formula, refer to 6.3.1, "DB2 CF Access Rate Calculation" on page 184.

$$\text{DB2 cache Write with Data} = \textit{BUFFER UPDATES} \times 0.5 = 469 \times 0.5 \approx 235 \frac{\textit{requests}}{\textit{second}}$$

$$\text{DB2 Page List Rate} = \frac{\textit{PAGES READ ASYNCHR.}}{24} = \frac{507}{24} \approx 21 \frac{\textit{requests}}{\textit{second}}$$

$$\text{DB2 Batch Unlock Rate} = (\textit{LOCK REQUESTS} - \textit{UNLOCK REQUESTS}) \times 0.1 \times 0.5 \Rightarrow$$

$$\text{DB2 Batch Unlock Rate} = (1348 - 852) \times 0.1 \times 0.5 \approx 25 \frac{\textit{requests}}{\textit{second}}$$

The lock and unlock request numbers come from Figure 68 on page 204 or Figure 69 on page 205. The 0.1 factor assumes batches of 10 for unlocking, and 0.5 is from our assumption of Type 1 indexes in the original setup. See the discussion in 2.2.4, "DB2 Locking Rate Calculations" on page 67 for a full justification.

Let us again return to Quick-Sizer. We are still in the window in Figure 70 on page 207 and we must now divide these previously calculated numbers into two parts in the ratio of two to one. This ratio comes from the workload split ratio for the two workload components:

1. Migrated off 9021-821 to 9672 (which is 40%)
2. Sharing in 9021-821 with 9672 (which is 20%)

First enter the numbers for Mgr: 9021-821 (migrated off 9021-821 to 9672):

- For DB2 cache Read without Data enter 91.
- For DB2 cache Read with Data enter 203.
- For DB2 cache Write with Data enter 157.
- For DB2 Page List Rate enter 14.
- For DB2 Batch Unlock Rate enter 17.

Then enter the numbers for Shr: 9021-821 (sharing in 9021-821 with 9672):

- For DB2 cache Read without Data enter 46.
- For DB2 cache Read with Data enter 102.
- For DB2 cache Write with Data enter 78.
- For DB2 Page List Rate enter 7.
- For DB2 Batch Unlock Rate enter 8.

When you have entered all the numbers, click on **Return**. You see the Output window in Figure 73 on page 212.

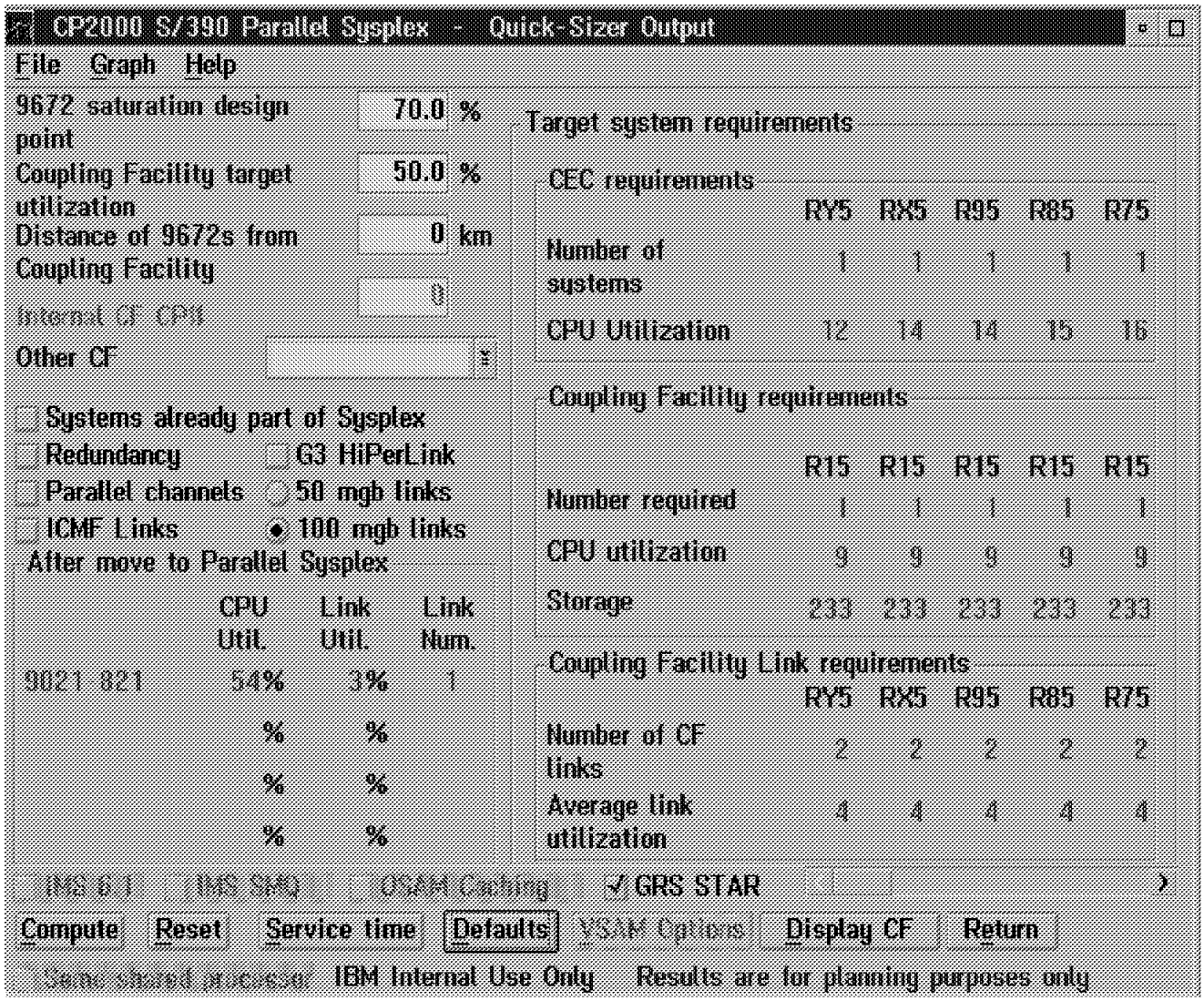


Figure 73. Quick-Sizer Output Window 2

Request the final computation by clicking **Compute**. Now the results are calculated. If you click the horizontal scrollbar at the bottom, you will see one column appear in pink. This is Quick-Sizer's way of suggesting the recommended choice.

Yellow and red background are used to show when, for example, the CECs exceed the saturation design points. Yellow is used when the CEC CP usage exceeds 70%, and red is used when it exceeds 80%. Note that on the right side of the output panel, the first and third subpanels are related. That is, the subpanel entitled **Coupling Facility Link requirements** refers to links on the corresponding 9672 CEC in the first subpanel entitled **CEC requirements**.

Click on **Display CF** for details of storage and CF usage estimates, as shown in Figure 74 on page 213. The Coupling Facility pulldown showing 9674-R15 allows you to change the type of Coupling Facility to obtain the estimates for other Coupling Facilities, and the 9672 Configuration pulldown showing 9672-RY5 allows you to change the 9672, although in general this should have little effect on CF usage.

Click on **OK** to return to the output panel.

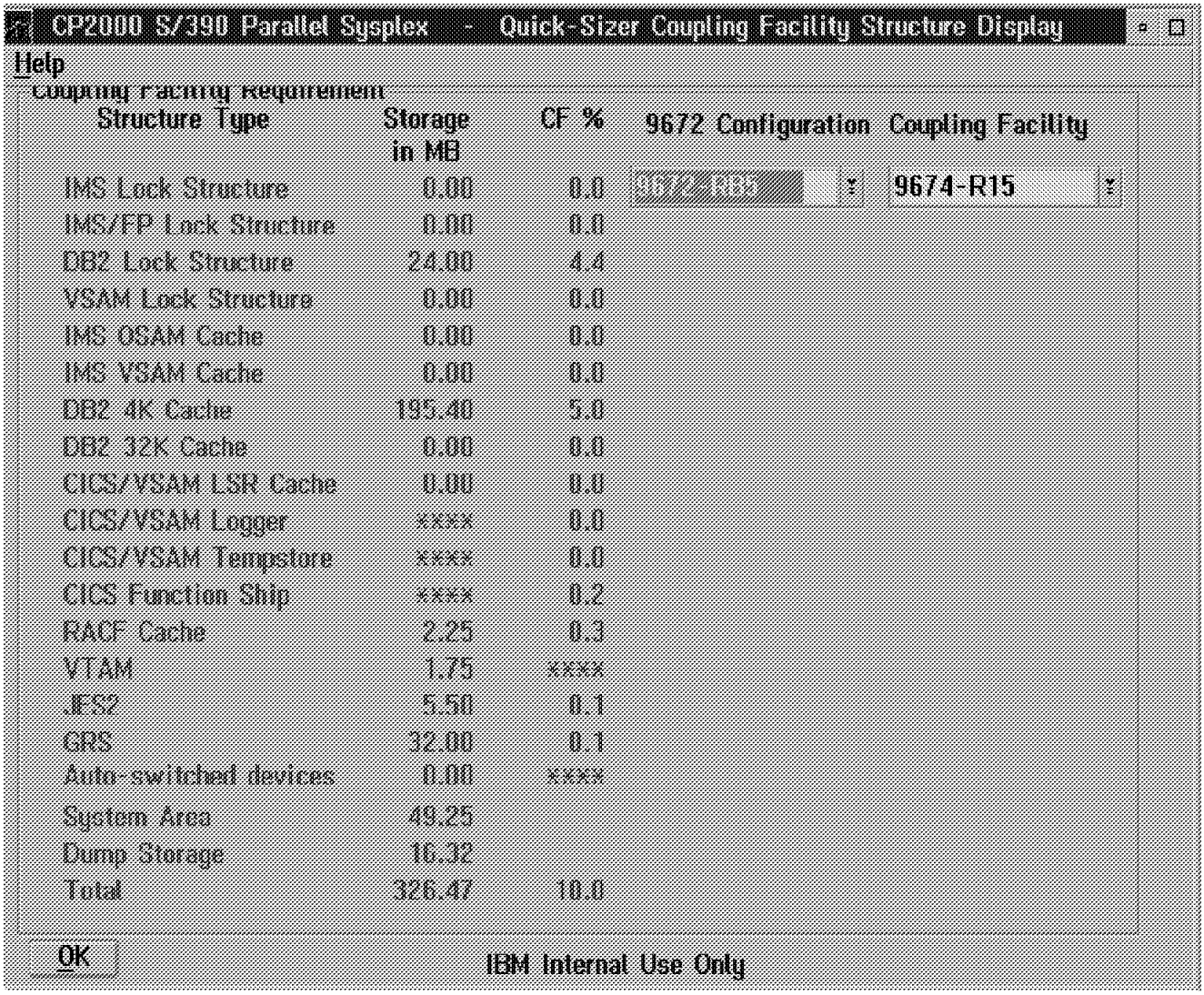


Figure 74. Quick-Sizer Display CF Window

You might want to examine the estimated service times for the CPU. When looking at the CPU service time, remember that:

- It does *not* include CPU queueing time, which is usually larger than the service time.
- It does *not* show the I/O time, which usually is at least three times as high as CPU service time.

Further, you can see data presented as graphs, as shown in Figure 75 on page 214. Click on **Graph** to show the CPU utilizations for the 9672 systems.

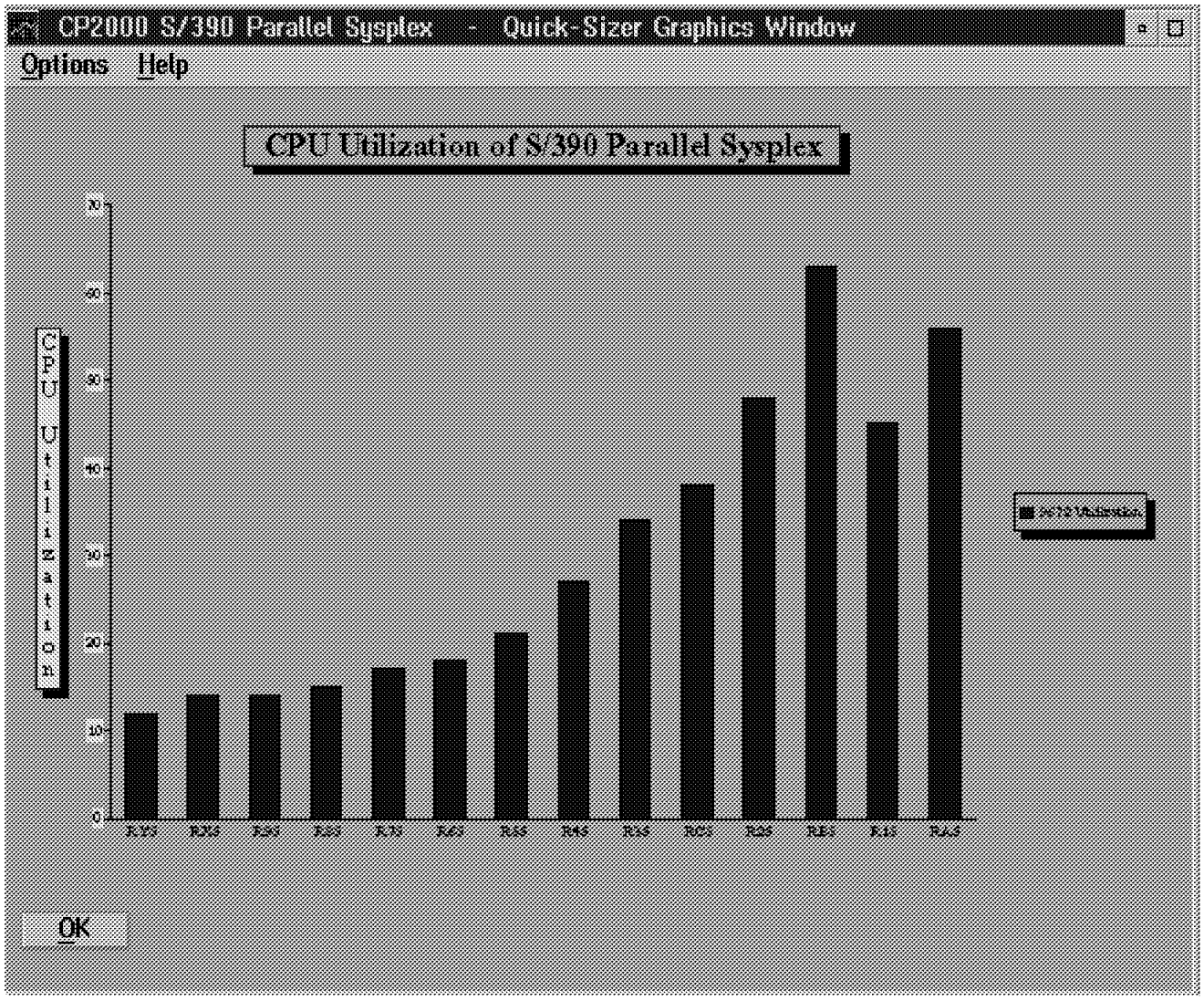


Figure 75. Quick-Sizer Processor Utilization Graph

This concludes our case study of a DB2 migration to a data sharing environment.

7.2 Case Study: Analyze Processor Speed

One of the main concerns in any system is response time. In this case study we look at *host* response time. By stripping away the network response time component, and only examining the host response time piece of the total response time. Then we go a step further: we primarily look at the CPU response time component of the host response time.

The CPU response time varies according to the following factors:

- The CPU processing power. The more powerful the CEC, the shorter response times you experience.
- The number of CPs in the CEC. The more CPs in the CEC, the shorter CPU queueing times you experience.
- The load of the CEC. The more loaded or used (CPU%) the CEC is, the longer CPU queueing times you experience.
- The priority or importance level of our application. The more applications that create load at a high priority, the longer queueing times you experience.

The last factor in the list, the application's priority level, sometimes creates surprisingly long response times. You should be aware of this phenomenon. Let us look at a small example where two workloads, Worklhi and Workllo, are first running at the *same* priority. This is depicted in Figure 76. In this example, the CPU response time is 4.3 times the CPU service time.

Both workloads have the same ratio for $\frac{response\ time}{service\ time}$ (rt/st).

Because the response time for the workload Worklhi does not meet its service level, and because for Workllo, the service level is met, Worklhi was placed at a higher dispatching priority level. Now Worklhi is running at a higher priority level: Here the ratio (rt/st = 2.3) is now better for Worklhi, and the service level is achieved. But we have now a very high ratio (rt/st = 19.2) for Workllo, and the service level here is not achieved.

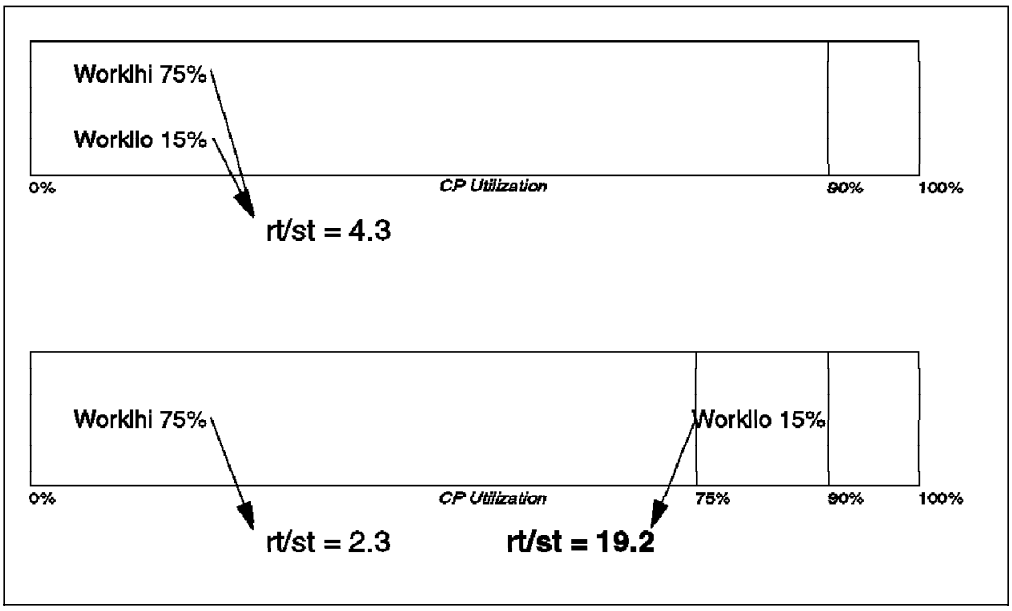


Figure 76. Workload Priority Influence on CPU Response Time: an Example

When migrating to a CEC with a higher processing power, there are usually no surprises because response times are shorter. Still, if the new CEC has less powerful CPs and your application is only able to use a single CPU, you have to be aware that CPU service time increases may increase your total response time. Additionally, your single CPU application may not fit the CEC at all if it requires more power than one CPU can provide. At a minimum you have to provide a high priority for your single CPU application.

CP2000 contains a model called Workload CPU Response Time Model for Single Image to calculate the CPU response time. It is based on the M/M/c queueing model (exponential multi-server model) and uses a “preemptive resume” priority queueing scheme. In general, this model applies very well both to OS/390 and user applications scenarios. The CP2000 model contains single processor application modeling as well.

In general, the CPU service time is usually less than one-third of the total host response time. The larger slice is often associated with I/O time. But if the CPU response time is extended due to heavy CPU load or a low priority of the application, then the CPU queueing part may be very long and play a significant role. It may create problems if it is not estimated and corrective actions taken in time. You have to be aware of this phenomenon.

7.2.1 Case Description

Let us suppose your installation has an IBM 9021-520 with three applications. These will be migrated to an IBM 9672-R32 or an IBM 9672-R24 system. Whether this system is part of a Parallel Sysplex or not plays no role in this case. The applications are running in the following priority scheme:

- Worklhi is the highest priority application, with seven transactions per second using 35% of the CEC.
- Worklmd is the medium-priority application, with eight transactions per second using 40% of the CEC.
- Workllo is the lowest priority application, with two transactions per second using 10% of the CEC.

All of them are CICS applications with several application regions. All the transactions use the DB2 databases.

In this scenario you want to investigate what will happen to the response times after the migration. We assume further that there are no problems today, even with the Workllo application.

7.2.2 Case Study Actions

The actions to be taken in CP2000 to analyze processor speed are as follows:

1. Start CP2000:
 - Click **OK** on the welcome panel.
2. Choose Single Image Processor Performance:
 - From CP2000 initial selection panel
3. On the Comparing Processors panel:

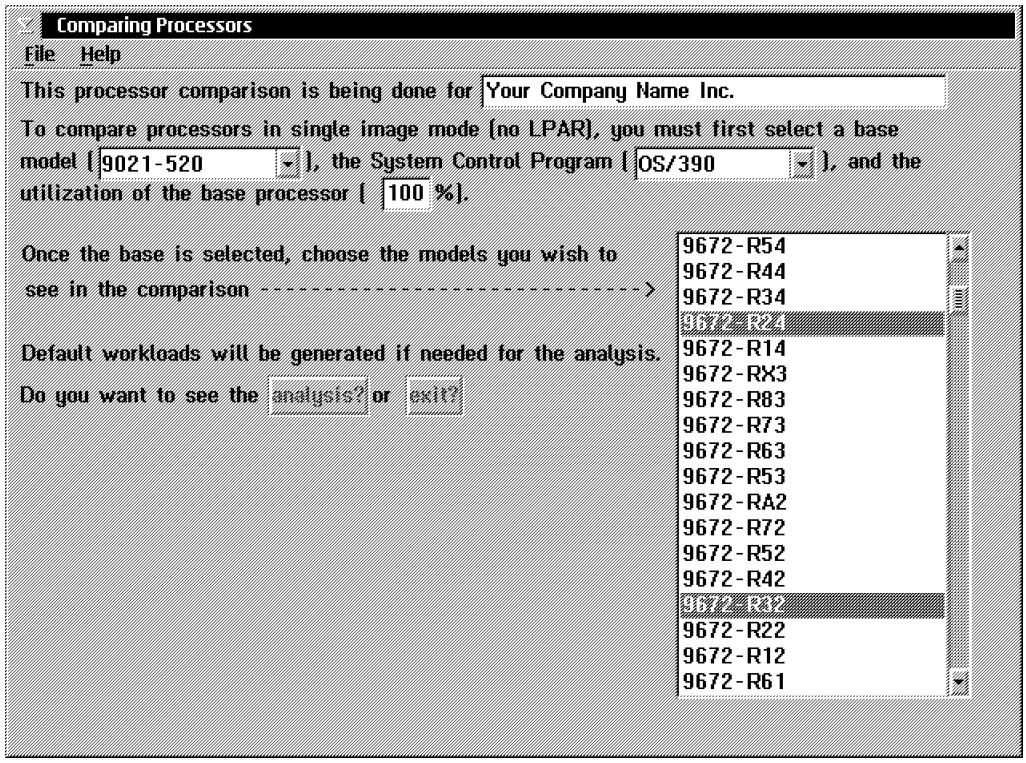


Figure 77. CP2000 Comparing Processors Panel

- Choose 9021-520 as the base processor from the pull-down.
- Choose 9672-R32 and 9672-R24 from the scrollable list of processors to be compared.
- Click on **analysis?**.

CP2000 generates default workloads and displays them on the CP2000 Workload Response Time Model for System Image Panel; see Figure 78 on page 218. To model our workloads we select a workload, which causes the Delete and Edit boxes to become enabled. We click on **Edit** for the first workload (which in the default is labelled System), and the CP2000 Workload CPU Response Time Model Add/Edit Workload panel appears. The only way to get a 35% utilization with 7 transactions per second is to give an appropriate MIPs/transaction figure. If we say that a 9021-520 is about 50 MIPs, then 35% at 7 transactions per second gives a path length per transaction of about 2.5 MIPs. Follow these steps:

1. Overtype the Base CPU% field with 35%. A Compute Trans Rate box appears.
2. Click the **Compute Trans Rate** box. At this stage, CP2000 is using its base MIPs/transaction value.
3. Overtype the MIPs box with 2.5. A Compute Rate box appears. Click this box.
4. Repeat step 3, adjusting MIPs up or down to get the correct transaction rate.
 - Remember, the first workload transaction rate is 7, the second workload transaction rate is 8, and the third workload transaction rate is 2.
 - Remember, CPU utilization is 35% for first workload, 40% for second workload, and 10% for the third workload.

- Remember to change #SYSTEM (under Description) to worklhi, change #INTERACTIVE to worklmd, and change #BATCH to workllo. Change workload type for worklhi, worklmd, and workllo to CICS/DB2.
- Change priority for worklhi to 254, worklmd to 100, and workllo to 1.

When the first workload is done, click on the **Next Workload** box and repeat the actions for this workload. When you have eventually edited all the workloads, click **OK** and you will get back to the Workload CPU Response Time Model panel as shown in Figure 78.

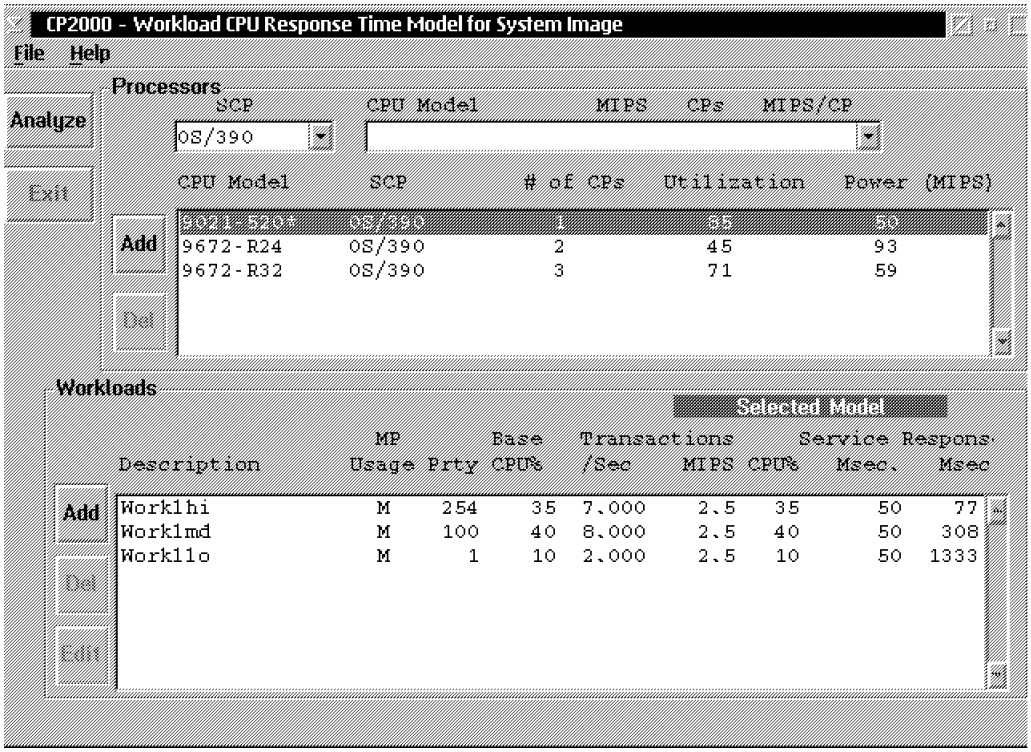


Figure 78. CP2000 Workload CPU Response Time Model for System Image

On this screen, click the **Analyze** box. The CP2000 Analyze Processor Speed panel is shown in Figure 79 on page 219.

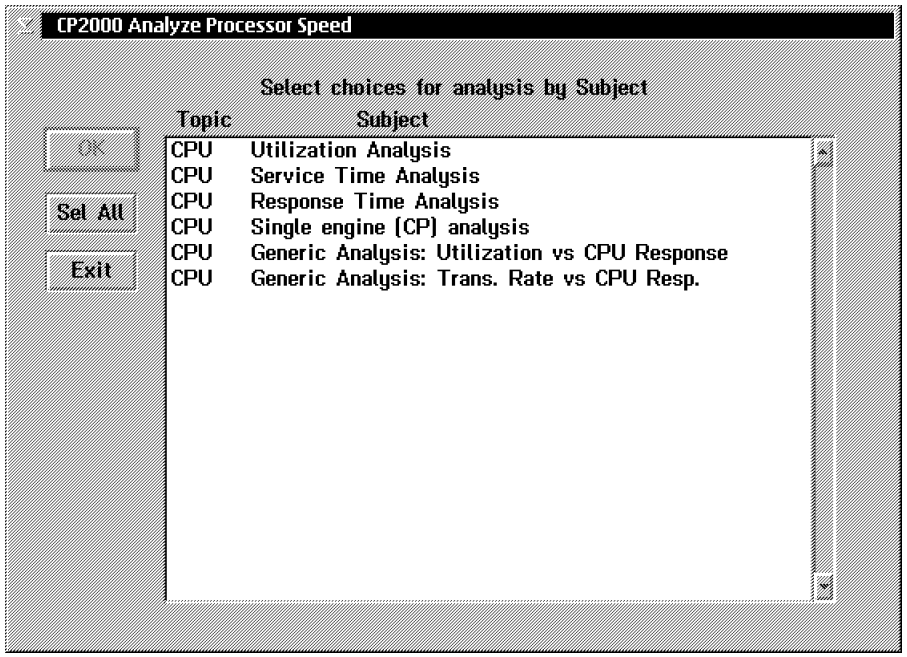


Figure 79. CP2000 Analyze Processor Speed

Click the **Sel All** box.

The processor utilization graph shown in Figure 80 is displayed.

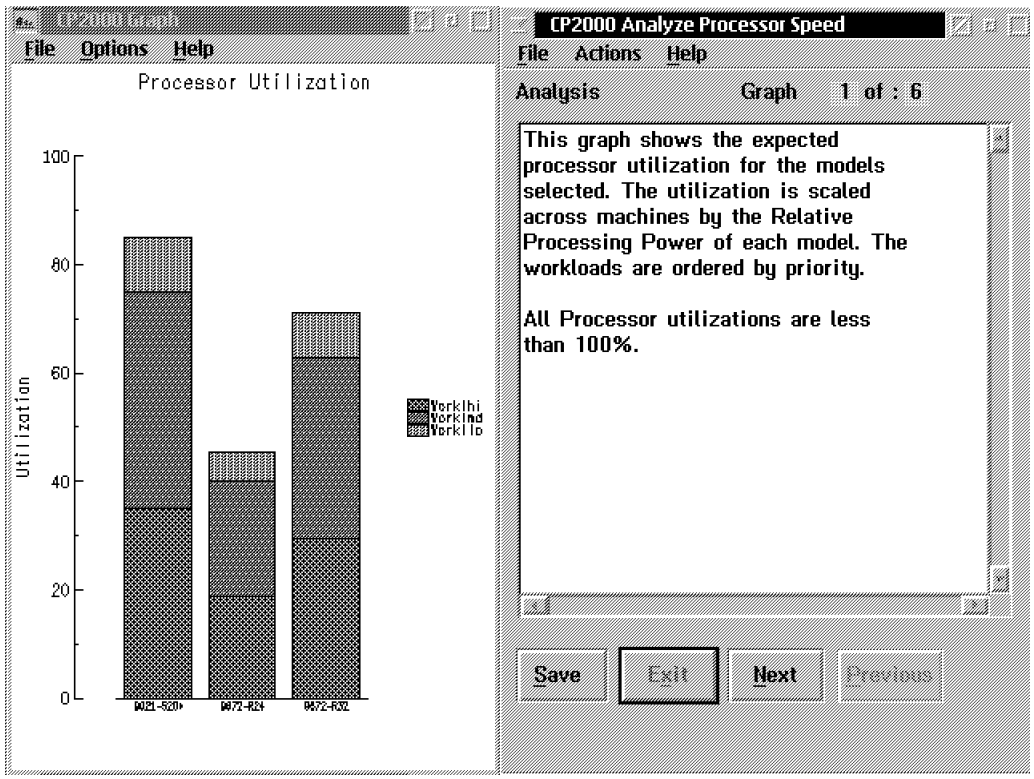


Figure 80. CP2000 Processor Utilization Graph

Click the **Next** box. The Processor Service Time graph shown in Figure 81 on page 220 is displayed.

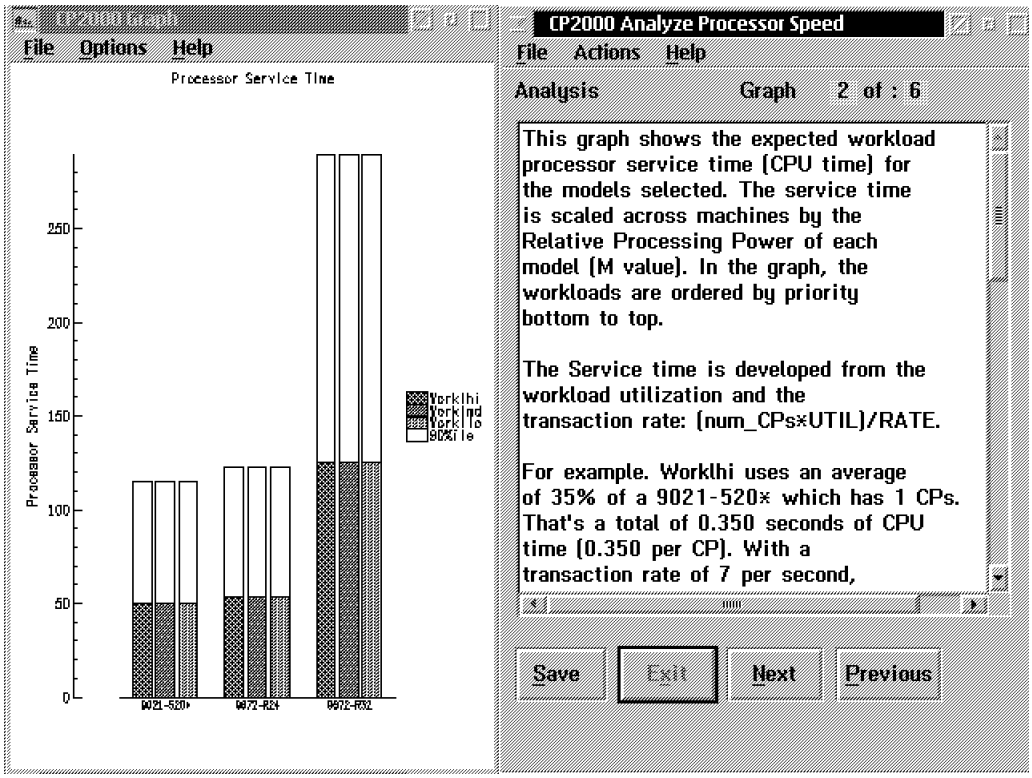


Figure 81. CP2000 Processor Service Time Graph

Click the **Next** box. The Processor Response Time graph shown in Figure 82 is displayed.

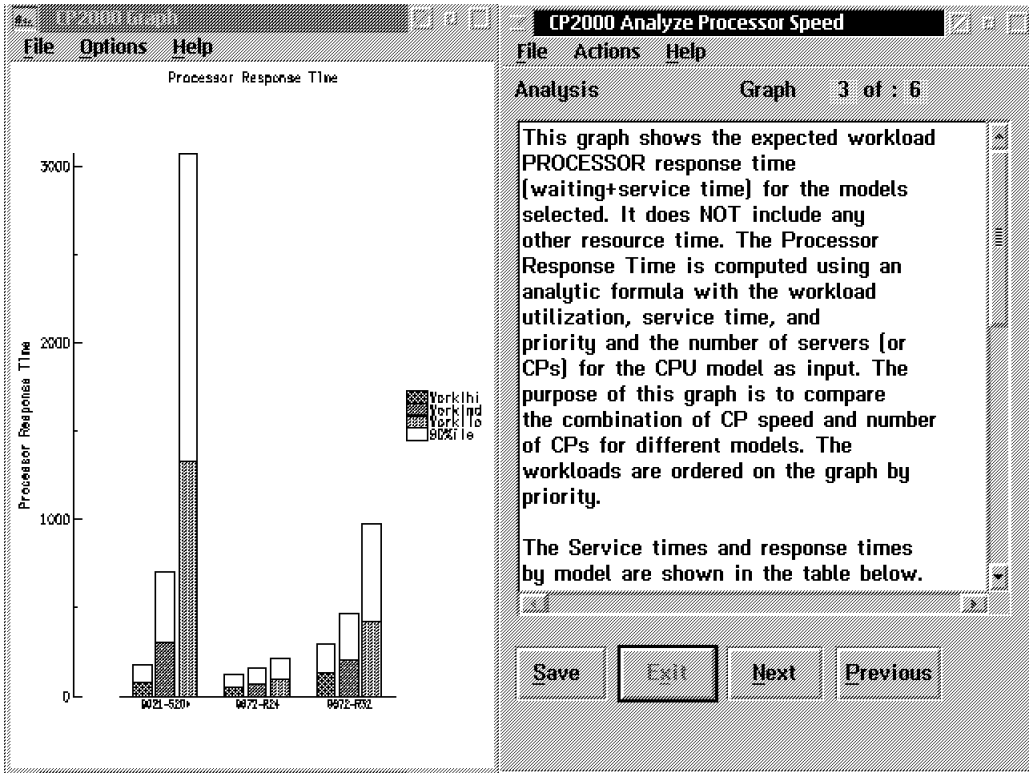


Figure 82. CP2000 Processor Response Time per Workload per CEC Graph

The response times (both the average and 90% percentile) for 9021-520, 9672-R32, and 9672-R24 are now shown. As you see, we can expect remarkably shorter response times from either 9672 system, not for the highest priority workload, but for the medium priority and especially for the low priority workload. This should not be surprising because the 9672 processors will be less heavily loaded.

Click the **Next** box. The Single CP Analysis graph shown in Figure 83 is displayed.

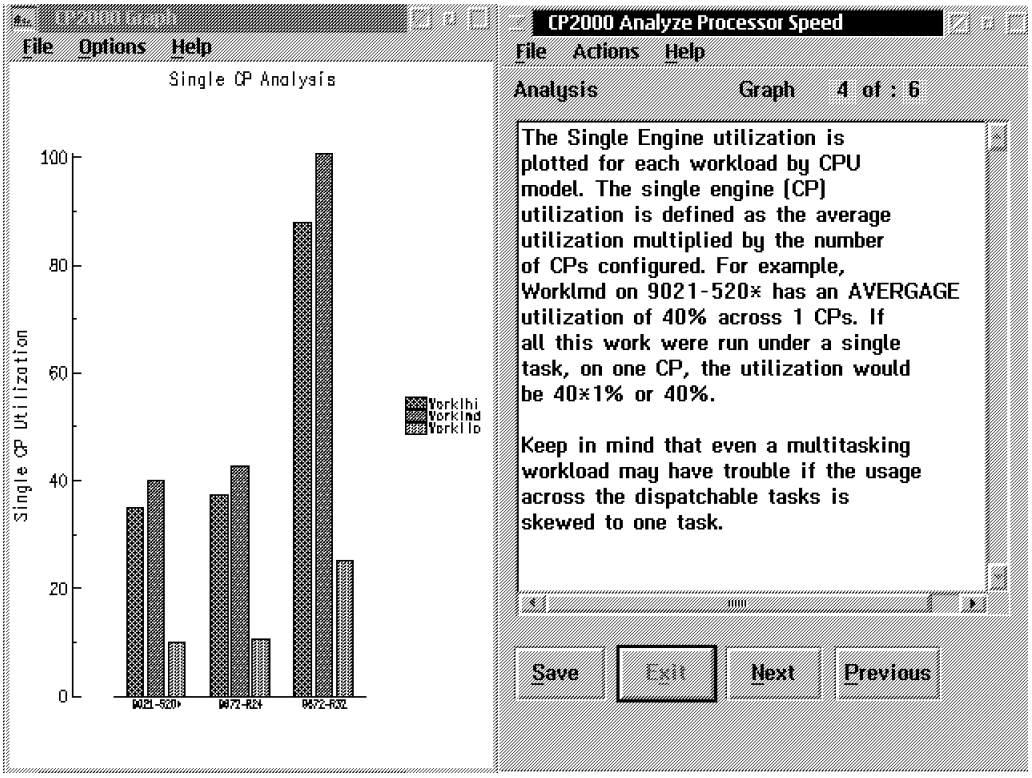


Figure 83. CP2000 Single CP Analysis Graph

We can see that both Worklhi and Workmd apparently use nearly a full CP on a 9672-R32. However, this is a CICS/DB2 workload, and in addition all uncaptured and system CPU time (such as JES, VTAM) has been amortized across the three workloads, so there should be no cause for concern. Click the **Next** box. The Processor Response Time graph shown in Figure 84 on page 222 is displayed.

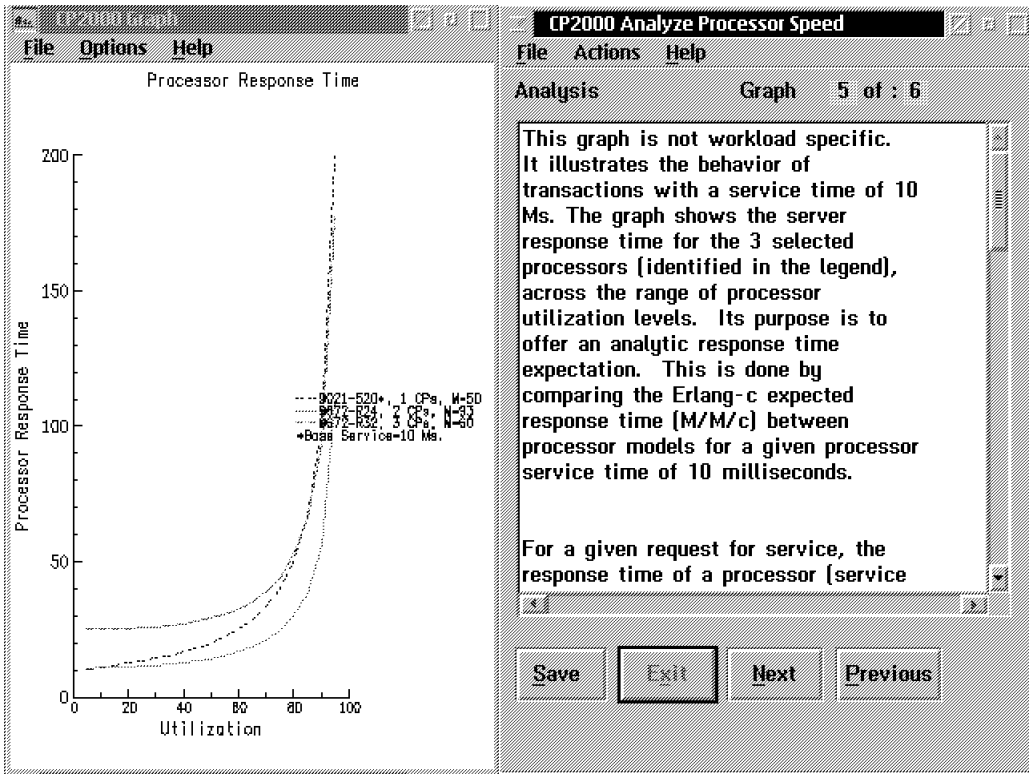


Figure 84. CP2000 Processor Response Time Graph versus Processor Utilization Graph

This differs from the earlier Processor Response Time graph. It is basically just an M/M/c plot of processor response time against processor utilization.

Click the **Next** box. Another Processor Response Time graph, shown in Figure 85 on page 223, is displayed.

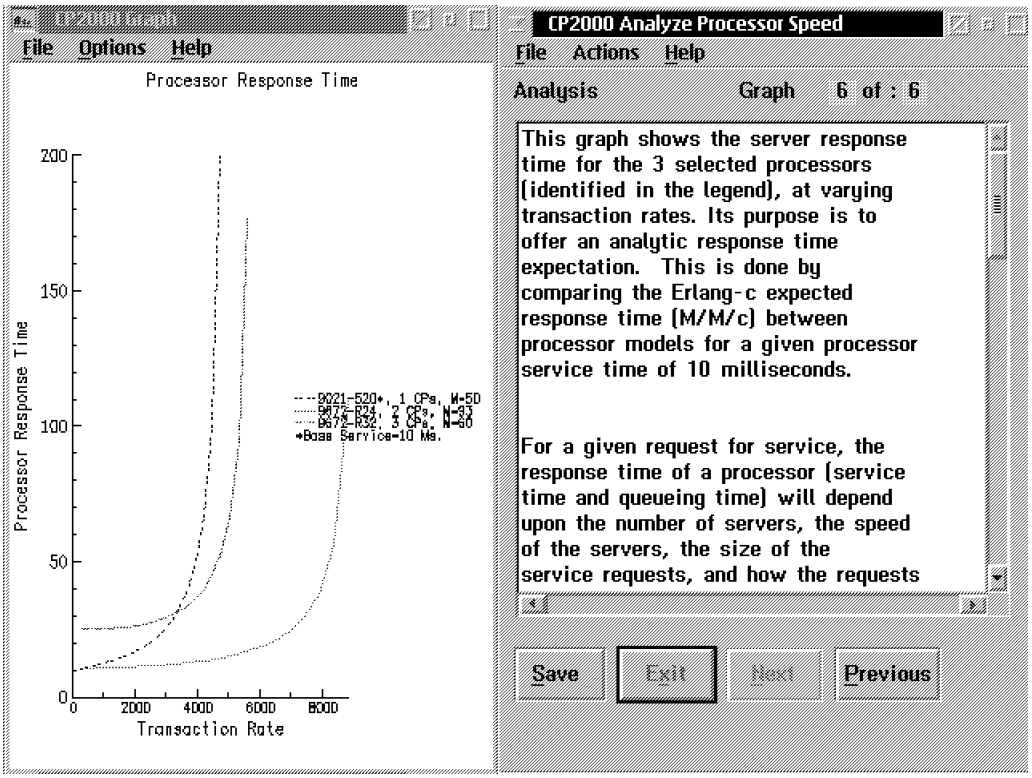


Figure 85. CP2000 Processor Response Time versus Transaction Rate Graph

This graph shows processor response time against transaction rate. It shows that the R24 can sustain response time at higher transaction rates than the 9021-520 or 9672-R32. This is not surprising, as the 9672-R24 has significantly greater capacity than either of the other two processors.

7.2.3 Assumptions and Restrictions for the M/M/c Queuing Model

The assumptions behind the calculations for the M/M/c queueing model are:

- The source size is infinite. For online applications this is generally true. In our case we have, say, over 200 terminal users. For batch jobs however this may *not always* be a valid assumption when only a few jobs are submitted.
- The system capacity is infinite. This is generally true, as there is often enough processor storage.
- The inter-arrival time distribution is exponential. This is generally true for online transactions. For batch jobs this may not fit. Batch jobs may be submitted, for example, at certain time points.
- The service time distribution is exponential. This is generally true.
- The number of servers equals the number of CPs. This is generally true. Exceptions to this rule are, for example:
 - Too few IMS MPP regions

 Make sure you have enough regions.
 - Too few CICS AOR regions

 Make sure you have enough regions.
 - Too few batch initiators for the specific job classes

Make sure you have enough regions.

- The applications are able to use as many CPs as available. This is often true. However there are exceptions, such as:
 - The application runs in one CICS region only and uses mainly one OS/390 task.
 - The application runs in one OS/390 address space only and uses mainly one OS/390 task. For example, IDMS is known to be mainly a single OS/390 task program.
 - We are looking at a single batch job only.

 For these cases, CP2000 provides an option called “Multiprocessing Usage.” It gives the capability of checking whether a single task application fits into the CEC.

The assumptions behind the calculations for the M/M/c priority queueing model are:

- A preemptive resume priority queueing scheme is assumed.

 For OS/390, this is generally true. Still, in the case of a large CEC (for example, a 9021) with LPs, LPAR overhead and OS/390 large system effects are minimized. This causes a slight delay in the dispatching of the next higher priority task, which means that a clean preemptive resume dispatching or interrupting scheme is not used. This effect is sometimes measurable. However, for rough response time planning purposes, there is no reason to worry about this effect.

- Average (mean) service times are equal.

 For online applications this is, in general, no restriction. Batch jobs generally use much more CPU than online transactions. Still, batch jobs are interrupted repeatedly by I/O interrupts, and as a consequence their CPU usage is split into small slices. Note that response time estimates (provided by the M/M/c priority queuing model) may be too pessimistic for batch jobs.

The CPU response time is just a part of the total host response time. Usually the CPU service time is less than one-third of the total host response time; I/O time is the dominant part. The following is the response time scenario *without* CPU queueing:

```
  CPU    I/O
!----!---------!  Total time = 3
  1       2
```

If the CPU response time is extended due to heavy CPU load or low priority of the application, then the CPU queueing part may be very long and play a significant role.

The following is the response time scenario with CPU queueing (compare this to the response time of Workllo in 3090-200J in our case study):

```
  CPU    I/O             CPU Queueing
!----!---------!-----------------------------------!  Total time = 15
  1       2                 12
```

The response time is extended, for example, from 0.3 seconds to 1.5 seconds for Workllo.

7.2.4 WLM Goal Mode

In WLM goal mode, you do not specify priorities. Instead you specify *goals*, either response time goals or velocity goals. This is an excellent facility, because you do not need to worry about priorities anymore: the only priorities you need to worry about are the “business priorities” that can now be specified when operating in goal mode.

OS/390 itself still uses dispatching priorities, but dynamically adjusts them to achieve the external goals set by you. This works fine without any additional definitions as long as you have specified reasonable goals, and as long as the load and the processing power in the CEC provide the possibility to achieve these goals.

If the specified goals are not achieved, OS/390 then respects the business importance definitions. The effect of the importance definitions is about the same as the compatibility mode setting for the priorities for the top, medium and low priority workloads. So in this situation, you can still compare the CPU response times for various importance levels. CP2000 provides a model for such calculations.

7.3 Case Study: CF Modelling

This case study uses the CF processing model, as described in Chapter 5, “A CF Processing Model” on page 143, to analyze the CF processing time components in a Parallel Sysplex environment and to evaluate the effects of changes to the CF hardware and in the link configuration. The CF processing model has been implemented in a spreadsheet which can be requested IBM internally from the RMFINFO tools disk:

```
TOOLS SENDTO BOEVM3 IIVTOOLS RMFINFO GET KGTV PACKAGE
```

7.3.1 CF Model Case Study: Environment

Figure 86 shows the Parallel Sysplex configuration which is used in the case study. The Parallel Sysplex consists of three 9672-RY4 systems, connected to two CFs, model 9674-R14.

The analysis looks at CEC1 (**1** in Figure 86), and models the CF processing time for requests sent to CF01. The dashed box **2** shows the scope of the model. Two LPs with OS/390 systems, J80 and J90, share the CEC. Two links are defined between CEC1 and CF01 and are shared between the two LPs. All input data required to model the processing time for CEC1 to CF01 is available from an RMF CF Activity report **3** .

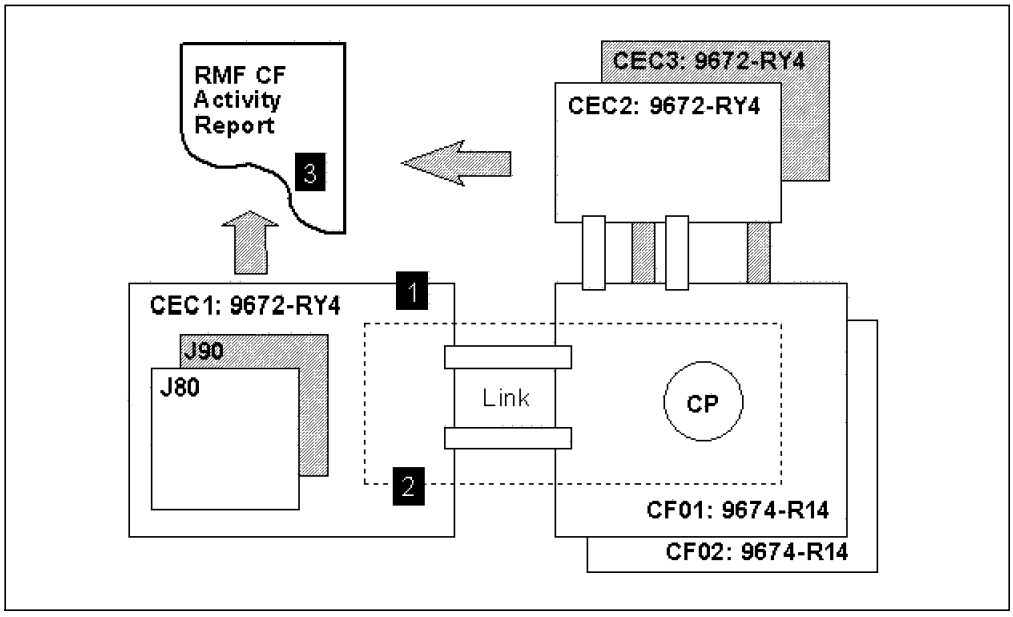


Figure 86. Case Study Environment

7.3.2 CF Model Case Study: Questions

The case study tries to model the environment described in 7.3.1, “CF Model Case Study: Environment” and helps to answer the following questions:

1. Where is most of the CF processing time spent for requests sent to the CF?
2. What does it mean to increase the number of requests for the current environment by 20%?
3. How does adding a CP for the CF affect the request processing time?
4. How will changing the number of links affect the request processing time?

The model can also answer more questions which are not discussed in this case study, such as:

- How does a different CEC influence the request processing time?
- Can faster links change the request processing time?

7.3.3 CF Model Spreadsheet

The spreadsheet that implements the model described in Chapter 5, "A CF Processing Model" on page 143 is easy to use. It is written in Lotus 1-2-3 Version 5 for Windows and can be used on all Windows and OS/2 platforms that have the International English version of Lotus 1-2-3 Version 5 or Lotus 97 installed.

The spreadsheet contains six worksheets that should be used by the user. These worksheets are color-coded (blue, red and green) and described as follows. (All other worksheets are used for calculations and should not be used).

Hint

It is recommended that you create a copy of the original spreadsheet before you perform a study. This guarantees that you still have a correct version if you accidentally destroy parts of the spreadsheet.

Intro Worksheet

The Intro worksheet contains a brief introduction and a disclaimer you should read before you use the spreadsheet.

Input_1 Worksheet

Use this worksheet to specify all input for systems sharing the same CEC and the same links. All information required on this worksheet is available on the Subchannel Activity section of the RMF CF Activity report (see Figure 87 on page 229). Information about the synchronous, asynchronous, and changed requests are required from the subchannel section of the RMF CF Activity report.

Input_2 Worksheet

This worksheet completes the input specifications. Besides the number of requests from the Subchannel Activity section, the following information is required:

- RMF interval length in seconds
- CEC model and number of CPs in CEC
- CF model, CF utilization and number of CF CPs
- Average number of requests processed by the CF
- Number of links, number of subchannels, and link speed

All this information is available from the usage summary section of the RMF CF Activity report (see Figure 88 on page 233).

In addition you have to specify the calibration factor for the model and a workload-dependent constant, which are both described in more detail later.

Model Worksheet

The Model worksheet consists of two parts (columns):

- On the right hand side is the *Model verification* part. It shows all calculated queueing, service, and processing times based on the input data provided in the previous worksheets. The spreadsheet implements the complex

queueing formula as described in 5.2.3.1, “Queueing Formulas for CF CP Processing Time Calculations” on page 149.

- On the left side you can define three scenarios to model the queueing, service, and processing time when you change the CF, CEC, and/or link configuration of your current environment. In addition, you can define an increase in the requests sent to the CF from the CEC you analyze, as well as from the other CECs connected to the CF.

Output_Graph Worksheet

This worksheet consists of two charts that graphically depict the effect of changes you made on the previous worksheets. We discuss all results of the case study using the chart in Figure 89 on page 236.

Output_QCF Worksheet

This worksheet contains one chart showing the effect of changing from a CF with *n* CPs to a CF with *n+1* CPs. This chart can be used for a quick analysis of changing the CF hardware, as long as the new CF model belongs to the same family or generation of CF models.

Macro Buttons

The worksheets “Input_2” and “Model” contain several macro buttons. These buttons help you use an estimated or calculated number for further processing: A **Reset** button on the Model worksheet helps you reset the scenarios to the input values of the current environment.

Note: The spreadsheet macro layout, number and name of worksheets, and number and name of fields, might be slightly different from the version described in this case study.

7.3.4 Defining the Input

All input required to model the request processing time is contained in the RMF CF Activity report. We will use a CF Activity report for a 15-minute interval for CF CF01.

7.3.4.1 Defining CEC Request Activity

The Subchannel Activity section of the RMF CF Activity report contains all information to define the request activity for the current CEC. Because two LPs (J80 and J90) share the same CEC and the same links, we must use the data for both systems in our model. Figure 87 on page 229 shows the subchannel section of the RMF CF Activity report with the two systems J80 and J90.

```
                                C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                                                PAGE  15
     OS/390                  SYSPLEX PRODPLEX           DATE 07/29/1997          INTERVAL 015.00.000
     REL. 02.04.00            RPT VERSION 2.4.0          TIME 10.00.00            CYCLE 01.000 SECONDS

------------------------------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = CF01
------------------------------------------------------------------------------------------------------------------------------------------------
                                                    SUBCHANNEL  ACTIVITY
------------------------------------------------------------------------------------------------------------------------------------------------
          # REQ                                  ----------- REQUESTS -----------   -------------- DELAYED REQUESTS --------------
 SYSTEM   TOTAL                  --BUSY--              #   -SERVICE TIME(MIC)-          #     % OF  ------ AVG TIME(MIC) ------
 NAME     AVG/SEC -- CONFIG --   -COUNTS-             REQ     AVG    STD_DEV          REQ    REQ   /DEL    STD_DEV     /ALL

 J80      804216  SCH GEN    4  PTH   33K  a SYNC    265615   180.5     305.9   SYNC    11K  4.0%  422.2       2375     16.7
           893.6  SCH USE  m 4  SCH   11K  b ASYNC   521950   962.1      2222   ASYNC   41K  7.8%   1783       5639    139.8
                  SCH MAX    4               c CHANGED    556  INCLUDED IN ASYNC  TOTAL   52K  6.5%
                  PTH      l 2               UNSUCC       0     0.0       0.0
 J90      806287  SCH GEN    4  PTH   32K  d SYNC    274072   184.8     390.1   SYNC    12K  4.3%  526.8       2701     22.9
           895.9  SCH USE    4  SCH   12K  e ASYNC   514528   962.2      2214   ASYNC   46K  8.9%   1623       4832    143.7
                  SCH MAX    4               f CHANGED    496  INCLUDED IN ASYNC  TOTAL   57K  7.3%
                  PTH        2               UNSUCC       0     0.0       0.0
```

Figure 87. Input from RMF Subchannel Activity Section

Worksheet Input_1 contains a table (see Table 17) in which we can fill in the synchronous and asynchronous requests and service times. The table contains entries for eight systems sharing the same CEC and links. In our example, we only have to use the two first rows. The information for system J80 is taken from the following rows of the subchannel section:

- **a** For synchronous requests, service time and standard deviation
- **b** For asynchronous requests, service time and standard deviation
- **c** For changed requests

In the same way we use the annotated rows **d**, **e** and **f** in Figure 87 to define system J90 in Table 17. Note that shaded areas in the following tables indicate spreadsheet fields that should not be overridden.

Table 17. Input Table (Worksheet Input_1)

System	Synchronous			Asynchronous			Changed
SMFid	Requests	SrvTime	StdDev	Requests	SrvTime	StdDev	Requests
J80	265615	181	306	521950	962	2222	556
J90	274072	185	390	514528	962	2214	496

Table 18 shows the information that is calculated based on your input in Table 17.

Table 18 (Page 1 of 2). Results (Worksheet Input_1)

Field Name	Field Value	Description
Number of LPs	2	
Total number of synchronous requests	539687	*35% of all requests*
Average service time for synchronous requests	183	*Microseconds*
Total number of asynchronous requests	1037530	*66% of all requests (include Changed Requests).*
Average service time for asynchronous requests	963	*Microseconds*
Total number of all requests	1577217	

Table 18 (Page 2 of 2). Results (Worksheet Input_1)		
Field Name	**Field Value**	**Description**
Average service time of all requests	695	*Microseconds*
Estimated calibration factor	5.15	*Value may be too high (see below)*

This information is also used and shown on the next worksheets.

Service Time used in Model

The model uses an average service time for all requests sent to the CF. The average service time calculated on worksheet Input_1 is a weighted average of the synchronous and asynchronous service times for all LPs sharing the CEC. Consequently, all predictions show an average service time for synchronous and asynchronous requests.

Estimated Calibration Factor

The worksheet estimates the calibration by dividing the weighted service time average by the weighted average for the standard deviation, and taking the result to the power of two (see 5.2.4, “Time Distributions and the k Factor” on page 154 for a detailed discussion on k factors). Experiments show that this number is too pessimistic and we recommend you use a value between the estimate and half the estimate.

7.3.4.2 Completing the Input

Worksheet Input_2 now asks for the remaining information to model the request queueing, service, and processing times. The input is divided into three categories:

1. General Input, including specification for the CEC
2. Input for the CF
3. Input for the links

The results of worksheet Input_2 are the CEC, CF and link service times, which are shown at the bottom of the sections. The following tables correspond to the spreadsheet. All output and description fields are shaded. They should never be changed on the spreadsheet. All references **g** ... **k** refer to the Usage Summary section of the RMF CF Activity report shown in Figure 88 on page 233 and the references **l** and **m** refer to the Subchannel Activity section of Figure 87 on page 229.

General Input:

Table 19. General and CEC Specific Input Data (Worksheet Input_2)		
Field Name	**Field Value**	**Description**
RMF Interval Length	900	*From* **g** *multiplied by 60*
CEC Model	9672-RY4	
Calculated CEC ITR	3.373	*Value extracted from PCR ITR table (see remark below).*
CEC ITR Ratio:	3.373	*We use the same value. You can change it if a more accurate number is available.*
CEC Service Time	85	*Microseconds (result of this input section).*

CEC ITRR

The spreadsheet macro contains an ITRR table for all current CEC models. The ITRR table was extracted from PCR (see A.1, “Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)” on page 249). A mixed workload is used and they are scaled to a 9672-R61. The CEC ITR value shown is the single engine power of the current CEC, which is required to calculate the CEC service time.

CF Input:

Table 20. CF Input Data (Worksheet Input_2)

Field Name	Field Value	Description
Calibration Factor	3.50	*We use a smaller value and not the estimation shown below.*
Estimated Factor	5.15	*This value has been estimated from input on worksheet Input_1 (see Table 18 on page 229).*
CF Model	9672-R14	*From* j*, we use the 9672 model that corresponds to our 9674-C04 model.*
Calculated CF ITR	2.65	*This value is not used later on.*
CF Utilization	49.8	*%, from* i
Number of CF CPs	1	*From* k
Avg Req/s	4260	*From* h
CF Service Time	117	*Microseconds; this is the calculated result of this section.*

CF ITRR

The CF ITRR is based on the same ITRR table contained in the spreadsheet, but the calculation is different. You will also find that the CF ITRR is lower than the ITR ratio for the CEC, even if they are the same models.

Link Input:

Table 21 (Page 1 of 2). Link Input Data (Worksheet Input_2)

Field Name	Field Value	Description
Calibration Factor	3.50	*We use a smaller value and not the estimation shown below.*
Estimated Factor	5.15	*This value was estimated from input on worksheet Input_1. (see Table 18 on page 229).*
Number of Links	2	*From* l
Number of Subchannels in Use	4	*From* m*, there are two subchannels per link and two link buffers per link.*
Link Speed	50	*MB/s; this installation uses 50 MB/s links*
Workload Factor	0.4	*This is a workload-dependent factor used to estimate the link utilization (see below). We use 0.4 to account for synchronous requests with and without data transfer.*
Link Buffer Utilization	30	*%, calculated from number of requests sent to the CF and number of link buffers in use.*

Table 21 (Page 2 of 2). Link Input Data (Worksheet Input_2)		
Field Name	**Field Value**	**Description**
Estimated Link Utilization	6.45	*%; using the workload factor, the link speed, and number of requests, we are able to estimate the link utilization.*
Link Utilization	6.45	*%; we use the estimated value. For G4 and G3 models with HiPerLinks we can get a more accurate number from the RMF Channel Activity report.*
Link Service Time	74	*Microseconds; this is the calculated result of this section.*

Workload Factor

The link utilization and the link service time depend on the amount of data transferred through the link. In order to account for the data transfer, our model uses a workload-dependent factor. This factor only influences the amount of data for synchronous requests. For asynchronous requests, we assume that there is always some data transfer. For workloads like DB2, which may send data with synchronous requests to or from the CF, we recommend a value of 0.7. For workloads like IMS V5, which do not send data with synchronous requests, we recommend a value of 0.1. For mixed workloads, a value of 0.4 is recommended.

```
                                C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                 g                                  PAGE   1
    OS/390                  SYSPLEX PRODPLEX           DATE 07/29/1997          INTERVAL 015.00.000
    REL. 02.04.00            RPT VERSION 2.4.0          TIME 10.00.00            CYCLE 01.000 SECONDS

------------------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = CF01
 TOTAL SAMPLES(AVG) =   896  (MAX) =   898  (MIN) =   893
------------------------------------------------------------------------------------------------------------------------------------
                                          COUPLING  FACILITY  USAGE  SUMMARY
------------------------------------------------------------------------------------------------------------------------------------
 STRUCTURE SUMMARY
------------------------------------------------------------------------------------------------------------------------------------

                                              % OF                 % OF      AVG    LST/DIR  DATA      LOCK     DIR
       STRUCTURE                    ALLOC     CF          #        ALL      REQ/   ENTRIES  ELEMENTS  ENTRIES  RECLAIMS
 TYPE  NAME             STATUS CHG  SIZE      STORAGE     REQ      REQ      SEC    TOT/CUR  TOT/CUR   TOT/CUR

 LIST  IXCLIST1         ACTIVE       20M        1.0%     2204K   57.5%    2449.1    4769      4755       N/A       N/A
                                                                                      18        39       N/A
 ...
                                    -------     -----   -------   ------   -------
           STRUCTURE TOTALS          378M       19.0%    3834K     100%    4259.5
                                                                      h

                                C O U P L I N G   F A C I L I T Y   A C T I V I T Y
                                                                                                   PAGE   2
    OS/390                  SYSPLEX PRODPLEX           DATE 07/29/1997          INTERVAL 015.00.000
    REL. 02.04.00            RPT VERSION 2.4.0          TIME 10.00.00            CYCLE 01.000 SECONDS

------------------------------------------------------------------------------------------------------------------------------------
 COUPLING FACILITY NAME = CF01
 TOTAL SAMPLES(AVG) =   896  (MAX) =   898  (MIN) =   893
------------------------------------------------------------------------------------------------------------------------------------
                                          COUPLING  FACILITY  USAGE  SUMMARY
------------------------------------------------------------------------------------------------------------------------------------

...

 PROCESSOR SUMMARY
------------------------------------------------------------------------------------------------------------------------------------
                                      j
 COUPLING FACILITY MODEL   9674      VERSION C04      CFLEVEL 3                        k
 AVERAGE CF UTILIZATION  (% BUSY)          49.8      LOGICAL PROCESSORS:   DEFINED   1    EFFECTIVE   1.0
                                      i
```

Figure 88. Input from RMF Usage Summary Section

7.3.5 Using the CF Spreadsheet Model

After completing the input, we verify the model and modify hardware-dependent parameters to demonstrate changes in the request processing time. The Model worksheet consists of two parts, the model verification on the left side and the part that allows for calculation of three different scenarios on the right side.

Based on the modelling result, we discuss the following three scenarios:

1. No configuration changes, but an increase of 20% in the number of requests from all systems
2. Same number of requests as in the current environment and change of CF model to a 9674-C04 with two CPs
3. Combination of one and two and we assume that we only have one link with two subchannels[12]:
 - 20% request increase
 - CF model: 9674-C04 with two CPs
 - 1 50MB/s link with two subchannels

To calculate the service and processing time, we use the formulas described in Chapter 5, "A CF Processing Model" on page 143. For convenience, here we repeat the formulas for the total service time:

$$S_{tot} = T_b\,(\text{send}) + T_c\,(\text{CF}) + S_{cf}\,(\text{CF}) + T_d\,(\text{receive}) + S_l\,(\text{Link}) + S_{cec}$$

and the processing time:

$$P_{tot} = S_{tot} + T_a$$

where:

T_a = Queue time to the link buffer
T_b = Queue time to the link
T_c = Queue time to the CF CPs
S_{cf} = CF service time
T_d = Queue time to the link
S_l = Link service time
S_{tot} = Total service time
P_{tot} = Total processing time

Table 22 shows all fields on the Model worksheet. There are no input fields in the model verification section. If you need to change anything, you can go to one of the input worksheets and change the corresponding input field. The table shows output fields shaded and input fields without shading.

Table 22 (Page 1 of 2). Model Output and Scenarios (Worksheet Model)

Field	Results	Scenario 1	Scenario 2	Scenario 3	Remark
CEC Model	9674-RY4	9674-RY4	9674-RY4	9674-RY4	*No change*
CEC ITR Ratio	3.37	3.37	3.37	3.37	
Used ITR Ratio:	n/a	3.37	3.37	3.37	*ITRR for scenarios*
CEC Service Time	85	85	85	85	*mics, = T_{cec}*
Sync Requests and % increase	539687	20%	0%	20%	*Increase applies for scenarios in %*
Async Requests and % increase	539687	20%	0%	20%	*Increase applies for scenarios in %*
CF Model	9672-R14	9672-R14	9672-R24	9672-R24	*New models for scenario 2 and 3*
CF ITR Ratio	2.65	2.65	2.56	2.56	*Calculated values*
Used ITR Ratio	n/a	2.65	2.56	2.56	*Values used in scenarios*
Number of CF CPs	1	1	2	2	*Values used in scenarios*
CF Calibration	3.50	3.50	3.50	3.50	*Calibration factor*
% Requests from other Systems	n/a	12%	0%	12%	*For 20% increase we need also more requests from the other two CECs*[13]
New Avg Req/s in CF	n/a	5121	4260	5121	

[12] This is not an example that can be used in reality, but it helps us demonstrate the effect of link modifications.

Table 22 (Page 2 of 2). Model Output and Scenarios (Worksheet Model)					
Field	**Results**	**Scenario 1**	**Scenario 2**	**Scenario 3**	**Remark**
Percent Requests increase in CF	n/a	20%	0%	20%	*Total increase in CF*
CF Utilization	49.8%	60%	26%	31%	
CF Service Time	117	117	121	121	*mics = S_{cf}(CF)*
CF Queueing Time	409	614	31	45	*mics, = T_c(CF)*
Link Calibration	3.50	3.50	3.50	3.50	
Number of Links	2	2	2	1	
Number of Link Buffers	4	4	4	2	
Link Speed	50	50	50	50	
Link Utilization	6.45%	7.74%	6.45%	15.47%	
Link Service Time	74	74	74	74	*mics, = S_l(Link)*
Link Queueing Time	18	22	18	47	*mics, = T_b(send) = T_d(receive)*
Service Time	720	933	346	419	*mics, = S_{tot}*
RMF Measured Service Time (Results) and Change in Service Time (Scenarios)	695	212	-347	-301	*mics, RMF Measurement and Modelling Result*
Modelling Error	4%	n/a	n/a	n/a	*RMF compared to S_{tot}*
Link Buffer Utilization	32%	49%	15%	44%	
Link Buffer Utilization	30%	n/a	n/a	n/a	*Based on RMF Measurements*
Link Buffer Queueing Time	58	349	2	352	*mics = T_a*
Total Processing Time	779	1282	348	771	*mics = P_{tot}*
Change to Processing Time	n/a	503	-431	-7	*mics*

[13] For a 20% increase of requests in the CF, we need 20% from our system and 12% from the other two CECs. These numbers are due to the different current usage of the CF and were found by iterative adjustments.

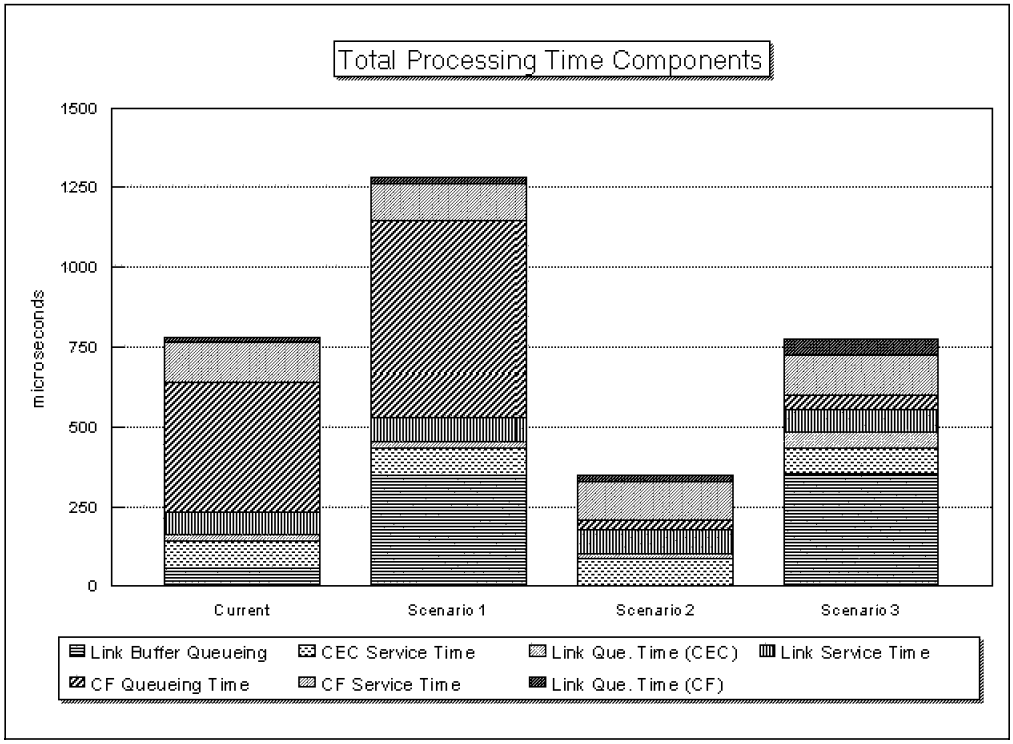


Figure 89. Case Study Results - Total Processing Time Components

Figure 89 and Figure 90 on page 237 display the modelling result graphically. The two charts are on worksheet Output_Graph. We use the graph in Figure 89 to discuss the result.

Current Environment

Based on our modelling, we clearly see in the current environment that the CF Queueing Time is the dominating factor in the request processing time. Since the CF queueing time is so high, the only real means for improving the processing time is to reduce the queues in the CF.

Scenario 1

Without changing anything else, but increasing the load by 20%, the situation in scenario 1 becomes much worse. We also see a very high link buffer queueing time, which is caused as a secondary effect from the CF queue.

One might conclude that adding links could help in this case - without knowing the original case - but the link buffer queueing time is only caused by the increased CF queueing time.

Scenario 2

Improving the CF queueing time by changing from a one-engine CF to a two-engine CF has a remarkable effect, as you can see in scenario 2. The CF queueing time has nearly vanished and the request processing time has improved significantly. Note that the processing time shown is an average between synchronous and asynchronous requests. We can expect to see very good service times for synchronous requests after changing the CF. We would also expect to see much less variation in service times, which means that the standard deviation in the RMF report will be reduced significantly (see Table 17 on page 229 and Figure 87 on page 229).

Scenario 3

Scenario 3 illustrates the effect of losing a link. This scenario has been added in addition to scenario 1 to demonstrate a real problem with the link. In this case, the CF queueing time improved, but we can see inflated link queueing times, and the dominating factor is the link buffer queueing time. We conclude that decreasing the number of links is not feasible.

Going One Step Further

Scenarios 1 and 3 demonstrate when adding a link helps and when it does not help. Taking scenario 1 one step further, our model would show that adding a link will reduce the link buffer queueing time from 349 microseconds to 26 microseconds. As a secondary effect, the CF queueing time would go up dramatically, because now more requests have to wait for the CF CP. This, in fact, would again increase the link buffer queueing time.

Attention

Add links only if there is no significant CF queueing time. Improving the link, as long as there is significant CF queueing, is not reflected correctly by the model and would provide optimistic and unrealistic results.

Figure 90 shows only the improvements in total service time (S_{tot}) and processing time (P_{tot}). This again helps to evaluate the effect of improving the links versus improving the CF configuration.

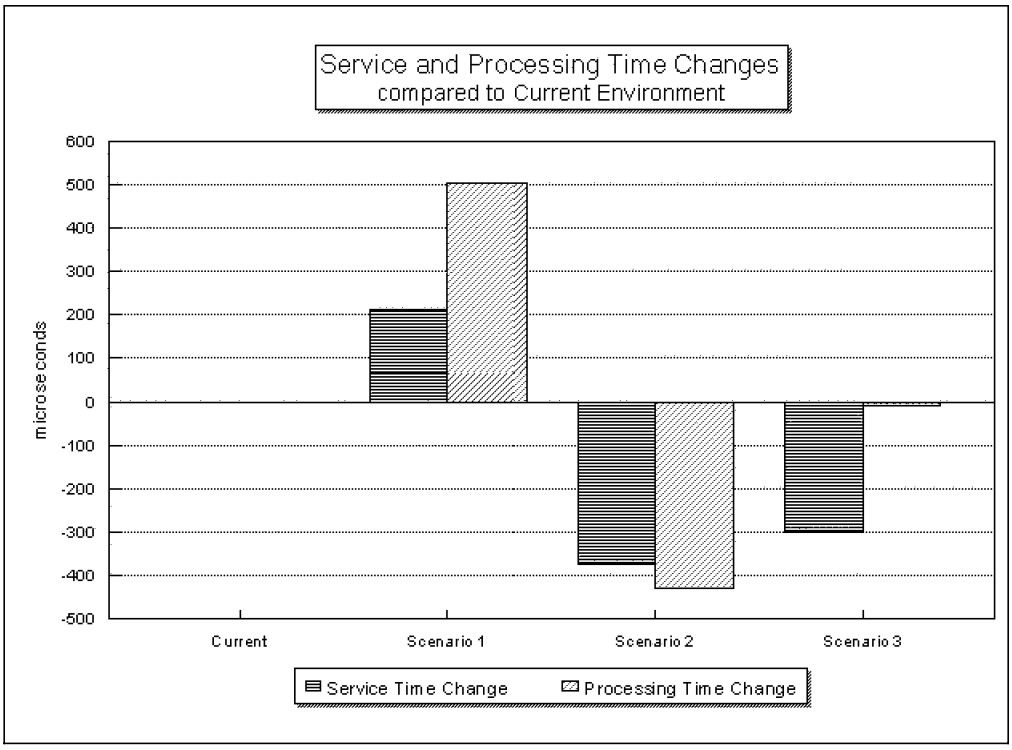


Figure 90. Case Study Results - Service and Processing Time Changes

7.3.6 CF Model Case Study: Answers

After completing our case study, we can now easily answer the questions from 7.3.2, “CF Model Case Study: Questions” on page 226:

1. Where is most of the processing time spent for requests sent to the CF?

 Answer: In the CF, shown by very high CF queueing times.

2. What does it mean to increase the number of requests for the current environment by 20%?

 Answer: That illustrates the worst case and shows that applications and end users will probably suffer in such a situation. It also means that the current environment does not have enough capacity for more intensive use by workloads.

3. Does adding a CP to the CF affect the request processing time?

 Answer: Yes, it solves all of our problems with CF responsiveness. Using scenario 3 without reducing the number of links from 2 to 1 will also easily show that this is a viable solution for future growth.

4. How will changing the number of links affect the request processing time?

 Answer: Adding more links does not really improve the CF processing time. We demonstrate with scenario 3 when this might help, and we show the pitfalls of our model with scenario 1.

We did not use the additional options the spreadsheet allows for, but we can probably easily answer the other questions as well:

- How does a different CEC influence the request processing time?

 Answer: In our case the CEC was not the critical factor. Therefore, improving the CEC would not dramatically change the results.

- Can faster links change the request processing time?

 Answer: Again no, because the problem was not due to the link. However, you may want to try faster links with scenario 3 to find out what happens.

7.3.7 CF Queuing Time Quick Graph

The spreadsheet contains one more chart on worksheet Output_QCF (see Figure 91 on page 239). This chart can be used as a quick overview of how having more engines in a CF can affect the CF queueing time, and from there, the total service time.

Note: This chart can only be used if you consider a model in the same family or generation of CF models. Figure 91 on page 239 is created based on three input parameters: the calibration factor, the CF service time, and the CF utilization. It may be safe to assume the calibration factor does not vary much, but the CF service time is different for each model. Within the same family of models, and without going from a one-engine CF to a ten-engine CF, the difference is small and the results still give a good approximation.

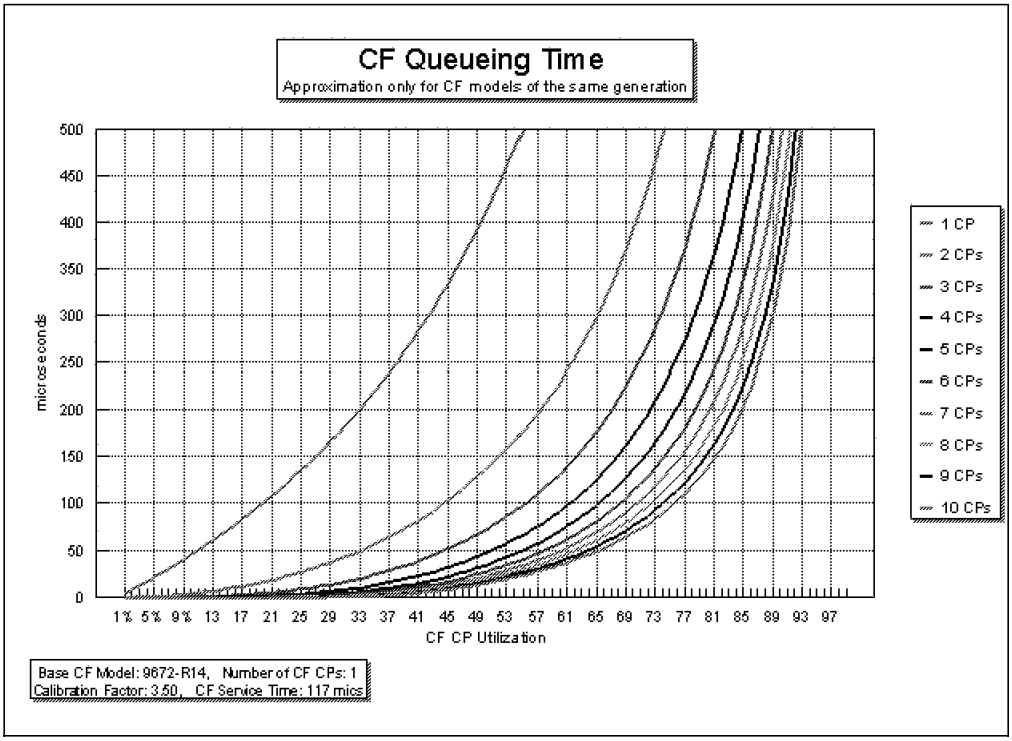


Figure 91. Case Study Results - CF Queuing Time

In Figure 91, we see that for a one-engine CF, the CF queueing time for 49% CF utilization is about 400 microseconds. Going from a one-engine CF to a two-engine CF down the 49% utilization line, we see that the queueing time drops from 400 microseconds to 130 microseconds.

Since we do not change the load, in this case the CF CPU utilization also drops to about one half of 49%. The complete calculation shows that the CF CP utilization is reduced from 49% to 26%. This again has a positive effect on the queueing time, which now goes down to 31 microseconds.

7.4 The OS/390 UNIX Services Sizer Tool

At a Glance

In this section we describe a prototype of an OS/390 UNIX Services sizer tool. Our examples, which are hypothetical are designed to show you how the tools may be used.

- One scenario shows how to use the OS/390 UNIX Services sizer tool, and describes its underlaying metrics.
- The other scenario shows an application running on two RISC/6000-59H machines which is moved to OS/390.

Note: At the time of writing the captured screens of the OS/390 UNIX Services Sizer Tool reflects an early prototype. The name was changed from MVS OpenEdition Sizer to OS/390 UNIX Services Sizer Tool.

7.5 How to Use the Tool

To use the OS/390 UNIX Services Sizer tool, do the following steps:

1. Select the source machine(s) (see Figure 94 on page 242).
2. Select the number of source machine(s) (see Figure 94 on page 242).
3. Select the type of application (see Figure 94 on page 242).
4. Select the utilization at which the source machine runs (see Figure 94 on page 242).
5. Add it to the compute board (Figure 95 on page 243).
6. Repeat steps 1 to 5 for every system you want to integrate on OS/390.
7. Calculate the target S/390 system.

This procedure is pretty simple but as described in 3.6, “Sizing for the OS/390 UNIX Services Environment” on page 115, it may be difficult to guess the input UNIX application CPU utilization. In addition, choosing the right workload type to resemble the input application may be error prone.. It may be even more difficult if you have a source machine that is not on the list, because you then have to search for an equivalent system which is on the list of “Vendors,” and/or “Machines.”

Hopefully this redbook will assist you to get past these difficulties.

7.5.1 Scenario One: Migrating an Existing UNIX Workload to S/390

After the logo has been displayed (see Figure 92 on page 241) you click **OK**.

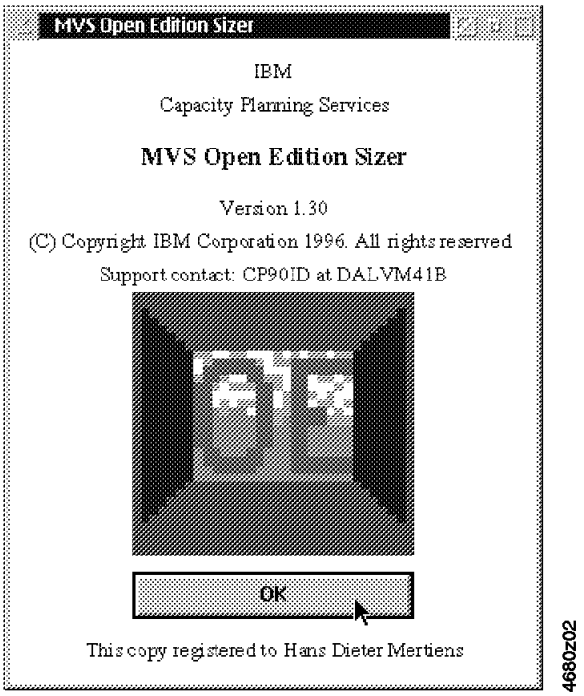


Figure 92. OS/390 UNIX Services Sizer Logo Screen

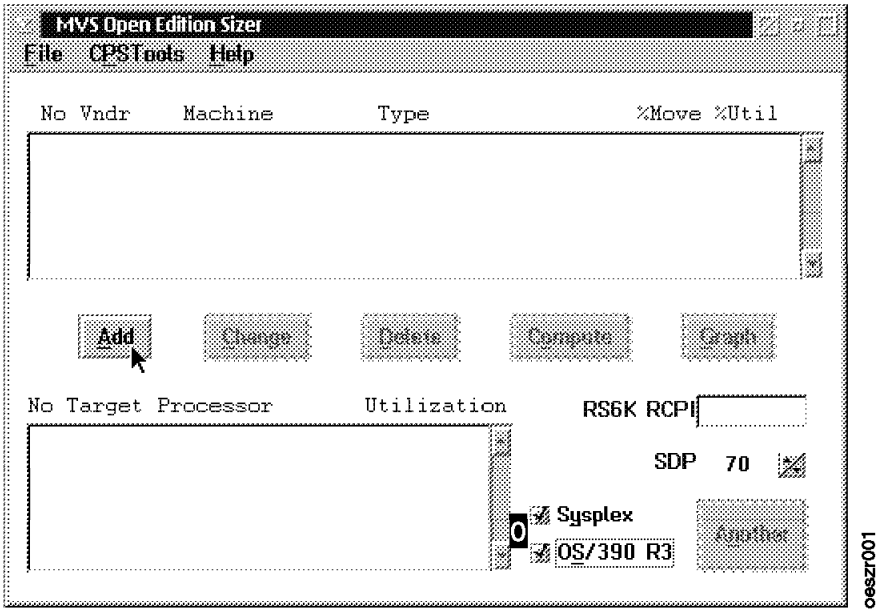


Figure 93. OS/390 UNIX Services Sizer Primary Panel

From the main screen (Figure 93), you can start with a new calculation using the **Add** button. The OS/390 UNIX Services Sizer tool also allows you to open a saved configuration that was defined earlier; you can do this using the **File** pulldown menu.

As you can see, we have applied both the **Sysplex** check mark and the **OS/390 R3** check mark (0). The OS/390 UNIX Services Sizer tool will do the following:

- Use several S/390 machines in a sysplex in case the application(s) to be moved do not fit on the largest S/390 system.
- Calculate according to the OS/390 UNIX Services performance enhancements implemented in OS/390 V1.3.

Continue by using the **Add** button, which leads us to a selection panel shown in Figure 94.

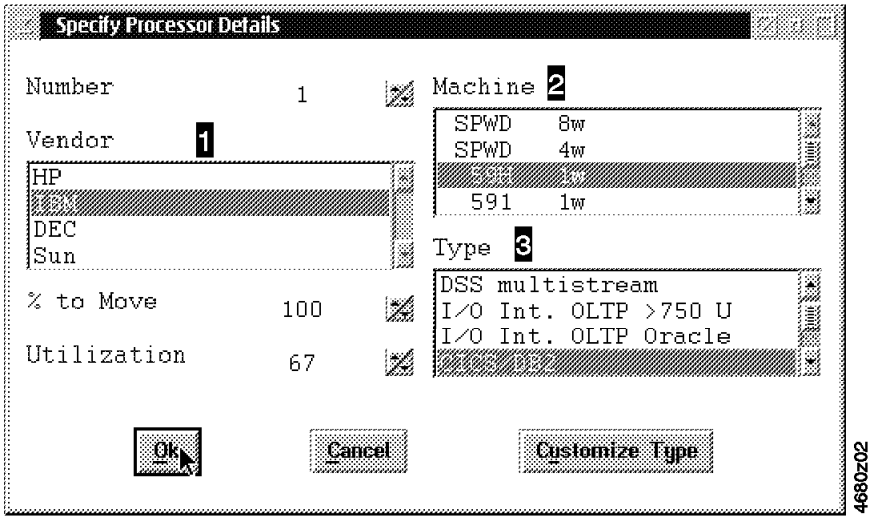


Figure 94. OS/390 UNIX Services Sizer Input Details

Here we can select, from three list boxes, which vendor and machine is currently used (**1** , **2**) and the classification or type of application (**3**).

Clicking on the **Ok**. button brings you back to the primary panel, as shown in Figure 95 on page 243 It now contains the source machine and application in the upper area (**4**).

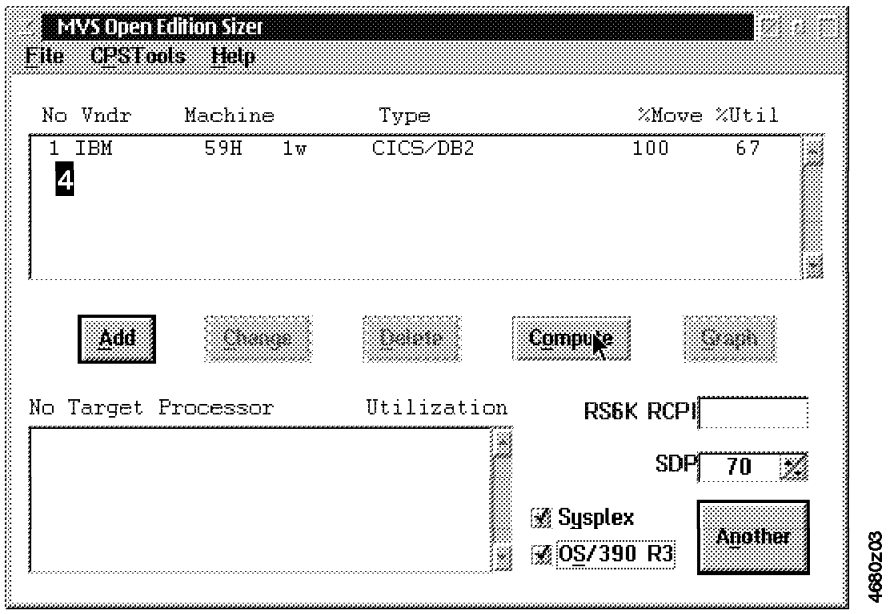


Figure 95. OS/390 UNIX Services Sizer Computation

Clicking the **Compute** button starts the calculation.

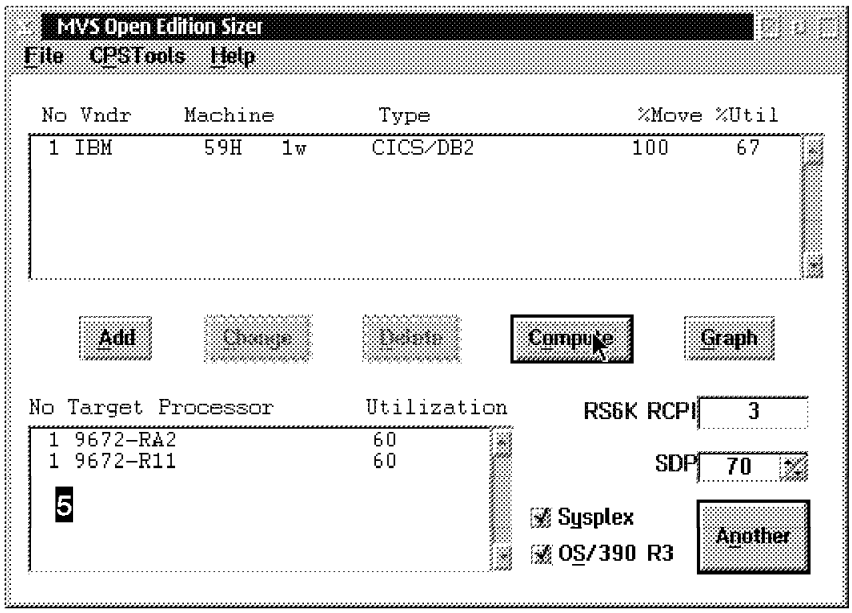


Figure 96. OS/390 UNIX Services Sizer OS/390 Result

After the computation is done, the OS/390 UNIX Services Sizer tool displays the result in the box labeled Target Processor (**5**).

The computation returns two S/390 systems. (Notice that later versions of the tool will only show 9672-Rx5 machines. Other machines may be analyzed by choosing the **Another** button.) The 9672-R11 is smaller than the R12 on which the model is based. This is due to the fact that we have taken into account the

OS/390 UNIX Services performance enhancements in OS/390 V1.3. Assuming an MVS/ESA V5.2 based system, we can verify the basic rules defined in 3.6, “Sizing for the OS/390 UNIX Services Environment” on page 115. Now we get exactly the base value (**6**) as shown in Figure 97.

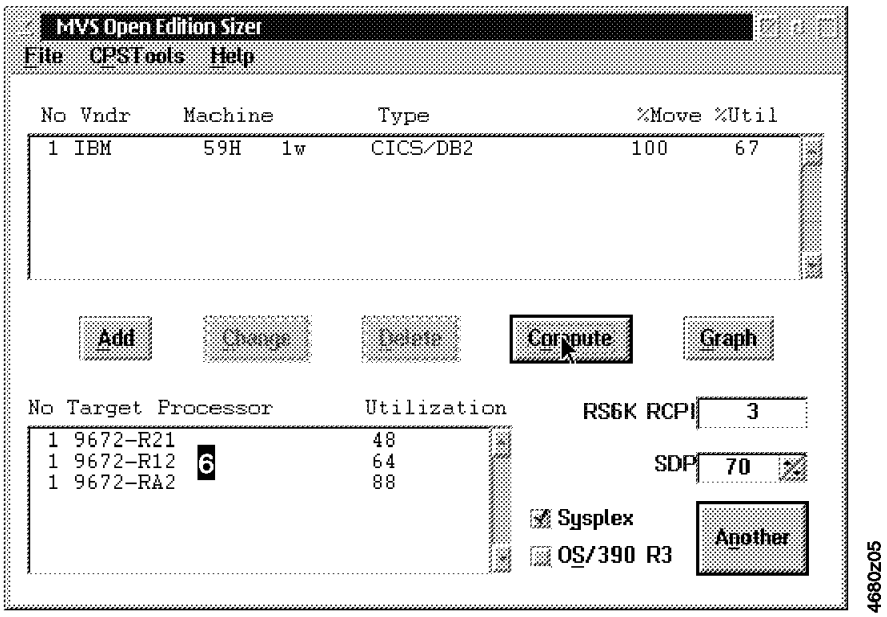


Figure 97. OS/390 UNIX Services Sizer MVS Result

In case you want to use a different S/390 system, you can use the **Another** button to go to a further selection list, which is shown in Figure 98 on page 245. Here you can select a different system according to your needs. Pressing the **OK** button will bring you back to the primary screen, which now contains the additionally selected system.

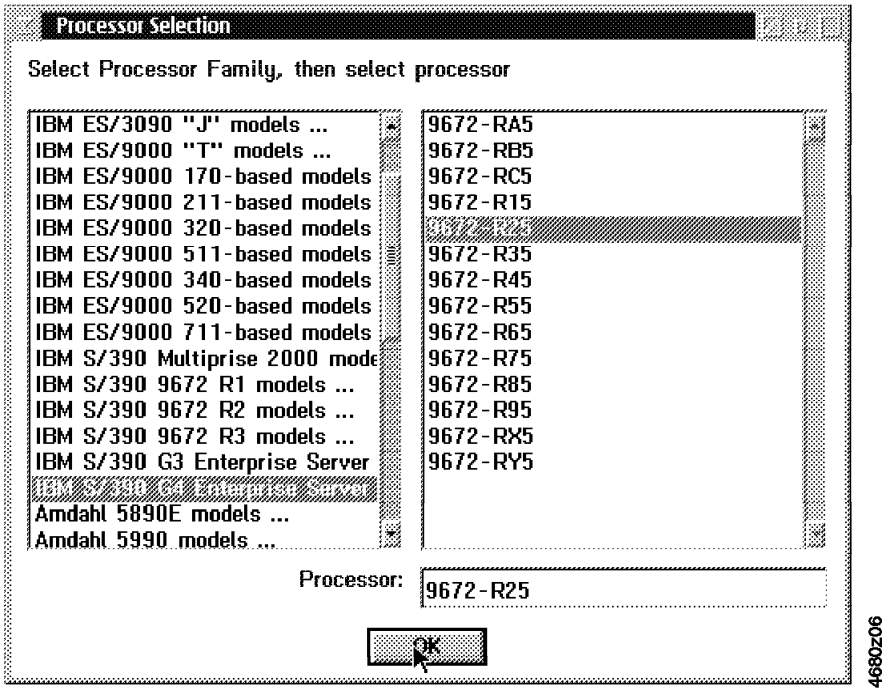


Figure 98. OS/390 UNIX Services Sizer Processor Selection

7.5.2 Scenario Two: Adding an Additional UNIX Workload to S/390

Scenario two shows the following:

1. How to modify an existing model, as previously calculated.
2. How to add another RISC/6000 system to the previous configuration.

We start from the screen as it is shown in Figure 97 on page 244. Using the **Add** button, leads us again to the selection of a source machine. We now add a load which is similar to an OE application (**7**), as shown in Figure 99.

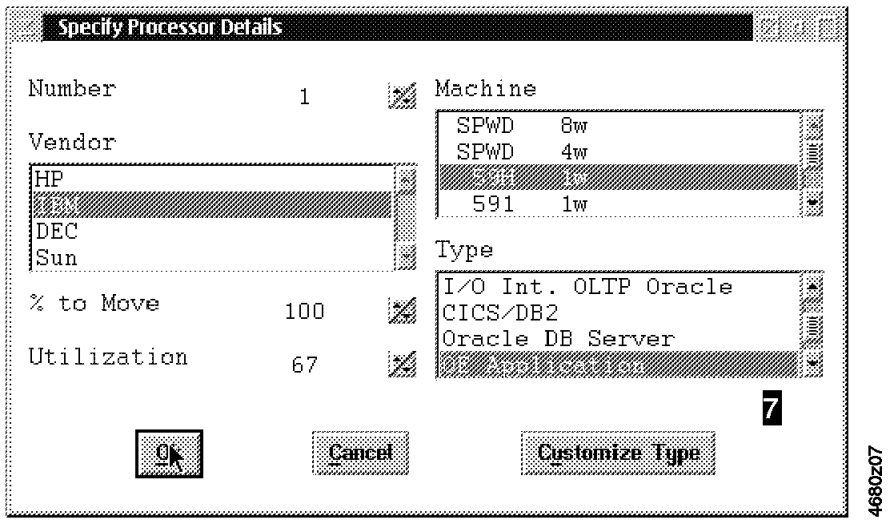


Figure 99. OS/390 UNIX Services Sizer 2nd Application

Again, after returning to the primary panel, we find the next input system added to the list (&8).

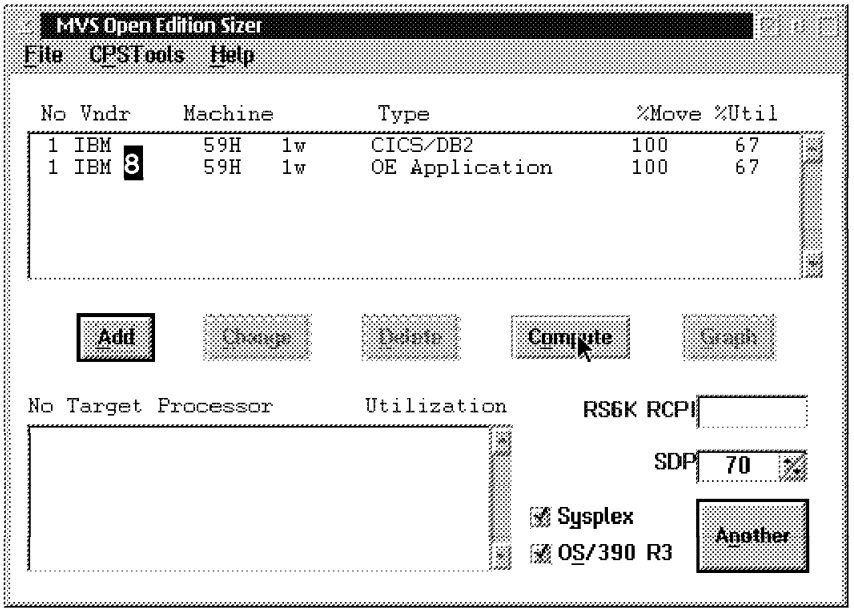


Figure 100. OS/390 UNIX Services Sizer 2nd Example

The computation then gives a new result (**10**) now displayed in Figure 101. The system that you previously calculated would now run at a higher utilization. Because we have defined a saturation design point of 70% (**9**), the OS/390 UNIX Services Sizer tool proposes to use a 9672-R21.

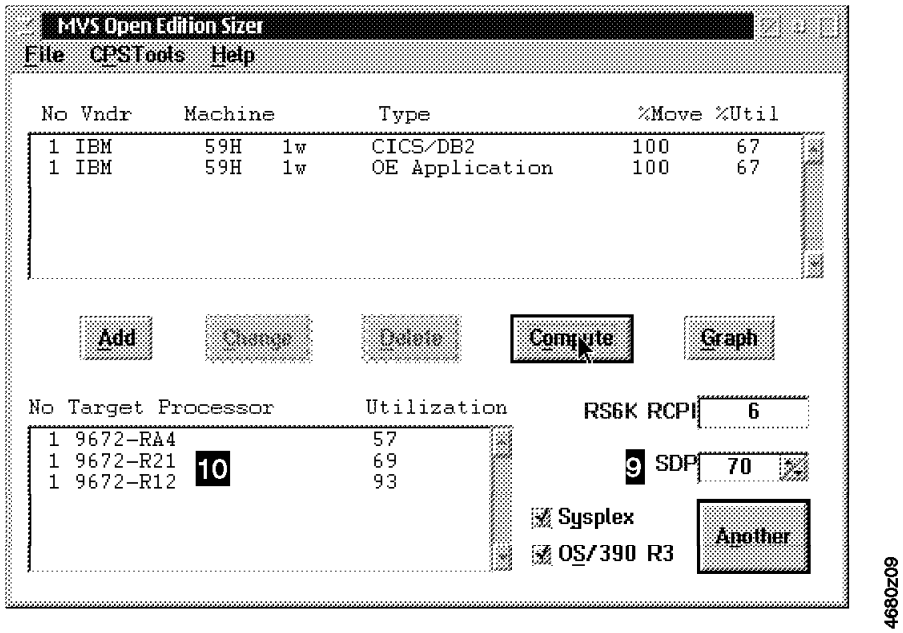


Figure 101. OS/390 UNIX Services Sizer Primary Panel

Appendix A. Capacity Planning-Related Tools and Services

At a Glance

This appendix briefly describes the capacity planning tools and services available.

It provides a table of capacity planning and related tools and services, as well as a catalog of tools, services and program products.

© Copyright IBM Corp. 1998

This appendix provides a brief description of capacity planning tools available for IBM representatives. Many of these tools are available for usage with customers; consult your IBM representative for more information. Various IBM services are described as well.

You can find information on these subjects in the following sections:

A.1, “Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)” on page 249

A.2, “LPAR/Capacity Estimator (LPAR/CE)” on page 253

A.3, “RMF Spreadsheet Reporter” on page 259

A.4, “Performance Management Services (PMOS)” on page 262

A tabular overview of other performance management and capacity planning-oriented tools are provided for your convenience in:

A.5, “Enterprise Server Capacity Planning Oriented Tools Table” on page 267

And finally, for your convenience, a catalog of many performance management and capacity planning-oriented tools is provided in:

A.6, “Enterprise Server Capacity Planning Tools Catalogue” on page 271

A.1 Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)

The Processor Capacity Reference (PCR) is a PC-based productivity tool, developed and maintained by IBM′s Capacity Planning Services organization (the developers of CP2000 and other tools). It is based on CEC capacity data provided by the S/390 division.

Note: PCR′s S/390 data is based on IBM′s "Large Systems Performance Reference" (LSPR) data. PCR is available to IBM representatives and approved IBM Business Partners.

A.1.1 PCR Introduction and Background

PCR is intended to provide capacity planning insight for System 390/370 CECs. Capacity data is included for both IBM and IBM-compatible CECs.

A.1.1.1 PCR and LSPR/PC

PCR uses the same data that is included in LSPR/PC. PCR has the functions of LSPR/PC and more. PCR is a CUA-compliant OS/2 application that is more intuitive to use than LSPR/PC. It requires OS/2 V3 or higher. Following are some PCR functions:

- Calculates the ITR ratio values for the different workloads
- Allows you to define your own mixed combination of workloads
- Allows you to use your favorite standards and values (RPP, MSU, MIPS or RIPS) for comparisons
- Allows new CECs to be easily added to PCR′s tables; their capacity is expressed relative to some CEC already in the tables
- Produces valid transaction response time curves based on either the CEC utilization or the transaction rate
- Produces tables for the ITR ratio (ITRR) values of specific interest
- Capacity of a single CP can be viewed for N-way CECs
- Estimates the life of a CEC model based on workload growth information for various workload components
- Computes the portion of a CEC required to support a workload consolidated from multiple CECs, based on current CEC utilization levels
- Provides analysis of specific LSPR workloads, allowing the user to adjust 4 metrics in order to assess the number of users or jobs that can be supported simultaneously
- Produces graphic and table output in formats that can be used as presentation materials or as handouts

It is the plan to add new functions over time; refer to *PCR Processor Capacity Reference, User′s Guide* for more information.

A.1.1.2 Basic PCR Analysis

A basic PCR analysis includes:

- Selecting the SCP for which the analysis is to be done.
- Selecting the workload environment.
- Selecting the base CEC, which will normally be a currently installed model that you are planning to replace. The capacity of the base can be scaled to any value or metric.
- Selecting one or more comparison CECs (models being considered as potential replacements).
- Documenting the results. PCR generates various tables and graphs, depicting CEC capacity relationships and related information. These tables and graphs can be output for use as presentation materials and handouts.
- Defining workload growth rates and determine where the capacity of CEC will be exceeded.
- Define workload consolidation scenarios and determine if they will fit on a planned CEC.

Input requirements for PCR are quite basic, including such items as the currently used CEC model (from which workload is to be moved), and the potential replacement CEC models. Alternatively, a target capacity expressed in terms relative to the current CEC can be used to determine potential replacement CECs. Capacity relationships can be scaled to match currently used information. Generally speaking, the use of PCR is simple and easy.

Note: PCR is written using floating point arithmetic and it requires approximately 4 MB of storage. We advise you to use a 486 or faster processor, or at least a floating point co-processor. Also be aware that PCR requires OS/2 V3 or higher.

A.1.1.3 Starting PCR?

After installing PCR, go to an OS/2 window, select **PCR directory** and type **PCR** or click on the PCR icon, and press Enter. Now the PCR welcome window comes up, and you press Enter to continue.

On the next window you select **S/390 - OS/390 and MVS**, then select **OS/390 level**, name your project, and select **OK**. The next window looks like the window in Figure 102 on page 251.

There is also a **PCRSETUP.CMD** included which, when used, will create the PCR icon for you and place it in the CPSTOOLS folder. It will also create the folder, if it does not already exist.

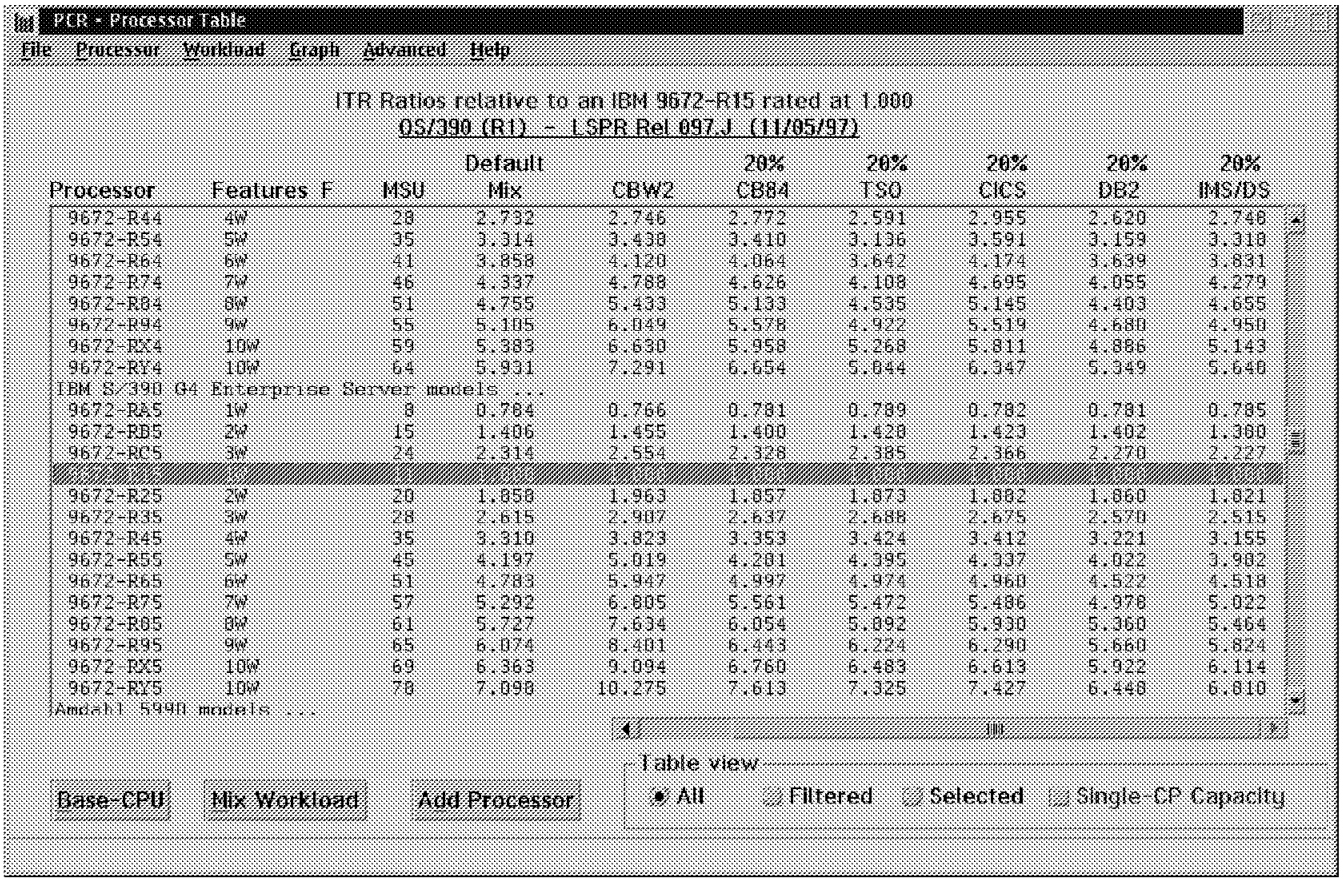


Figure 102. PCR Processor Table Window

The window in Figure 102 shows the ITR ratios table using default selections:

- For the Base CEC (9672-R15)
- Its ITR ratio (equal to 1.000)
- Workload mix (20% of each of CB84 (batch), TSO, CICS, DB2 and IMS/DS)

All the previous fields can be changed. From here on it is very easy to continue: just click or select the required topic and specify your requirements.

A.1.1.4 PCR Further Information

The *PCR Processor Capacity Reference, User's Guide* gives you complete guidance. The use of PCR is easy and there are useful help windows in PCR, so the need for the user's guide is minimal.

The user should refer to *Large Systems Performance Reference (LSPR)*, SC28-1187 for specific discussions about LSPR data and its use.

A.1.2 LSPR/PC

LSPR/PC is a productivity tool designed to make relative capacity assessments for System 370/390 architecture CECs running various OS/390, VM, and VSE workload environments. It is a PC-based application implemented as a compiled spreadsheet. The tool is based on IBM's "Large Systems Performance Reference (LSPR)" Internal Throughput Rate (ITR) data, and on the CEC comparison method described in the *Large Systems Performance Reference*

(LSPR), SC28-1187. LSPR/PC is available for IBM representatives and can be used with customers.

LSPR/PC results are presented instantaneously, using graphics, with supporting documentation available. Graphs may be captured for use in presentations. The LSPRPC package is available on MKTTOOLS. After you have downloaded the appropriate files to your PC, you can start for example, LSPR/PC MVS/ESA by entering: `RUN MVS/ESA`. Once the target LSPR/PC spreadsheet has been loaded, macro execution begins automatically. Menus are presented at the top of the screen, from which you will direct LSPR/PC activities. Choose a menu item by moving the cursor to the desired item and pressing Enter.

A.1.2.1 LSPR/PC Functional Overview

There are three primary functional areas implemented in LSPR/PC. *TABLE* allows you to deal with the entire table as an entity, while *SELECT* and *ANALYZE* provide a more detailed analysis with some graphic output. If you select *TABLE:* the CEC table is presented, with the capacity of each CEC for each workload given relative to that of the preassigned base. The table, as presented, can also be output to a printer, as a file, or as a spreadsheet. You can modify the table to:

- Change the base CEC
- Assign a scaling factor to the base CEC
- Define a mixed workload

With the **ANALYZE** and **SELECT** functions you can determine a required capacity for a consolidation or you can compute simultaneous users or jobs supported. You can also customize your view of the CEC or the workload table:

- CEC comparison

 Refine the number of CECs being analyzed to a base CEC and up to four comparison CECs. The base CEC can be scaled to any value.

- Workload definition

 Select a specific workload to be active during analysis. A mixed workload, defined in business-units, can be selected.

- Growth rate

 Enter, for example, a five year growth scenario, either for the workload as a single entity, or for a mixed workload, by business-unit.

The *LSPR/PC User's Guide* is included in your received package. There is a context-sensitive help facility available.

A.2 LPAR/Capacity Estimator (LPAR/CE)

The LPAR Capacity Estimator provides capacity planning for CECs with the PR/SM feature and running in LPAR-mode. LPAR/CE is available to IBM representatives and approved IBM Business Partners.

A.2.1 Introduction to LPAR/CE

LPAR/CE assumes a scenario where multiple independent basic-mode CEC environments are to be merged on a single CEC as a separately defined LP. LPAR/CE has its basis in actual LPAR benchmark measurements. The tool has been validated by modeling each of those LPAR test cases, and comparing the resulting data to that measured.

A.2.1.1 Intention of LPAR/CE

LPAR/CE is intended for approximating the capacity potential of a PR/SM CEC running in LPAR-mode. LPAR/CE is *not* intended to be used for:

- LPAR performance expectations
- LPAR overhead factors
- ITR ratios between basic-mode CECs
- Specific ITR values

Once a model has been built, you have a characterization for each partition′s workload, and for the LPAR host CEC in general. This information can then be used to evaluate the following:

- Various configuration alternatives for an existing LPAR host CEC
- The effect of changing the LPAR host CEC to another CEC
- Effect of workload growth in one or more partitions
- The inclusion of an additional workload from separate basic-mode CEC into a new partition on an existing LPAR CEC
- The effect of migrating a partition′s workload from one LPAR CEC to a different LPAR CEC.

A.2.2 LPAR/CE Implementation and Operation

LPAR/CE was developed on an IBM PS/2, using the spreadsheet capabilities afforded by Lotus 1-2-3 Release 3. A DOS[14] run-time module, used for executing the compiled spreadsheet, is supplied with LPAR/CE. Therefore, there are no special software requirements to use the tool.

A.2.2.1 LPAR/CE Input and Summary Panels

The panel shown in Figure 103 on page 254 contains all necessary input fields to describe several possible LPAR configurations. For details, refer to the *LPAR/CE User′s Guide*, which is part of the LPARCE package on CPSTOOLS.

[14] DOS Release 3 or higher, or DOS window under IBM′s OS/2, or under Microsoft′s Windows

```
{1} Input and Summary Panel                                    LPAR/CE

   Environments to be combined under LPAR       <--- optional input --->
    Current    (LSPR)    (LSPR)    CPU    LP    Wkload   Cstore    DASD
LP  Processor   SCP      Workload  Busy  Mode   Growth   Estore   CHPIDs
--  ---------  --------  --------  ----  ----   ------   ------   ------
 1. 9021-720   OS/390    CICS       65%  S8              512       48
 2. 3090-200J  OS/390    IMS        45%  S4      10.0%   512       64
 3. 9021-520   OS/390    CB84       55%  S3              256       22
 4. 4381-92E   OS/390    TSO        70%  S2      25.0%   128       18
 5.
 6.
 7.
 8.
 9.
10.

 Target PR/SM Processor:  9672-RX3     Total capacity needed:     84.8%
 -- (Adjusted for LUE) ---------------->  Average utilization:     86.6%
  Estimate of host LPAR-mode capacity / host B-mode capacity:     93.1%
```

Figure 103. LPAR/CE Input and Summary Panel. (Input fields are highlighted)

Figure 103 shows the primary input fields for LPAR/CE, which define your current CEC environment(s). With this panel you also define the target LPAR environment (in this case, a 9672-RX3).

LPAR/CE calculates the total capacity needed on the target CEC. We show you some of the output LPAR/CE calculates in the following topic.

A.2.2.2 LPAR/CE Output and Summary Analysis Panels

Sample LPAR/CE panels are shown in Figure 104, Figure 105 on page 255, and Figure 106 on page 256. LPAR/CE generated output includes:

- Real CP resources required for the LPs
- Suggested weights for the LPs
- LPAR cost

```
 {2} Real CP Resource Usage Panel                                  LPAR/CE

  0 CPs dedicated (0% of total) --->  Maximum capacity needed:      n/a
 10 CPs shared (100% of total) ---->  Average capacity needed:     84.8%
 10 CPs total (LCP:RCP ratio = 1.70)
                                          Pct of     Pct of
                                       10 Real CPs  Avail. Suggested
    9672-RX3 LPAR Environment Defined  Usable Avail. Needed  Weights
    ---------------------------------  ------ ------ ------ ---------
 1. SHR  8-W LP, running  OS/390  CICS   80.0% 100.0%  52.7%   1= 622
 2. SHR  4-W LP, running  OS/390  IMS    40.0% 100.0%  13.6%   2= 160
 3. SHR  3-W LP, running  OS/390  CB84   30.0% 100.0%  15.2%   3= 179
 4. SHR  2-W LP, running  OS/390  TSO    20.0% 100.0%   3.3%   4=  39
 5.
 6.
 7.
 8.
 9.
10.
```

Figure 104. LPAR/CE Real CP Resource Usage Panel

```
 {3} Partition Capacity Panel                                      LPAR/CE

Compare LPAR host's partition capacity to that of the
  currently installed basic-mode processor.                 Rel.  (Shared)
                                                 Apparent  Weight Adjusted
    9672-RX3 Partition / Current CPU             Capacity Assumed Capacity
    -------------------------------------       -------- ------- --------
 1. SHR  8-W LP : 9021-720,  OS/390  CICS          98.6%    622     76.6%
 2. SHR  4-W LP : 3090-200J,  OS/390  IMS         132.8%    160     53.0%
 3. SHR  3-W LP : 9021-520,  OS/390  CB84         108.4%    179     64.8%
 4. SHR  2-W LP : 4381-92E,  OS/390  TSO          422.6%     39     82.5%
 5.
 6.
 7.
 8.
 9.
10.
                                                                   -------
    9672-RX3 LPAR capacity / total of 4 current CPUs                 69.6%
```

Figure 105. LPAR/CE Partition Capacity Panel

```
 {5} Analysis Summary Panel                                        LPAR/CE

LPAR/CE capacity metrics below are influenced by the workload quantity
 and type as defined to each configured partition in Panel-1.  Any
 effects of user-entered weights on workload mix are also considered.

 Target LPAR host processor  . . . . . . . . . . . . . 9672-RX3 (10 CPs)
 LPAR host family  . . . . . . . . . . . . . . . . . .   9672-R3
 Partitions defined  . . . . . . . . . . . 0-DED  4-SHR  . . .        4
 Logical CPs defined . . . . . . . . . . .    0     17   . . .       17
 Partitions benefiting from MVS alternate WAIT-STATE mgmt  . .        4
 Logical CPs defined per Real CP . . . . . . . . . . . . . . .    1.700
 LPAR management time (-). . . . . . . . . . . . . . . . . . .    -4.87%

 Estimate of host LPAR-mode capacity / host B-mode capacity  .    93.11%
 Net LPAR cost(-) or benefit(+)  . . . . . . . . . . . . . . .    -6.89%


 LPAR-mode capacity / B-mode 3090-180J rated at   1.00 . . . .     6.81
```

Figure 106. LPAR/CE Analysis Summary

A.2.3 LPAR/CE Considerations for Parallel Sysplex

Parallel Sysplex implementation involves the use of a CF. A CF runs Coupling Facility Control Code (CFCC) and is usually implemented using a stand-alone 9674 CF. Alternatively, CFCC can be run in a logical partition under LPAR on 9672 processors, 9021 711-based processors, and 9121 511-based processors. CFCC running in a logical partition can use real CF links, or it can use the Integrated Coupling Migration Facility (ICMF) to emulate the coupling links. CFCC using real coupling links is referred to as a CF partition. CFCC using the ICMF facility is referred to as a CF/I partition.

There are special considerations for CF partitions when using LPAR/CE to assist in capacity planning for these activities.

A.2.3.1 CFCC Using Real Coupling Links

CFCC in a CF partition is not interrupt driven; it runs in continuous execution, or “active wait.” This means that a CF partition will utilize CP resources up to the limit of its logical processor capacity. Therefore, dedicated CPs are highly recommended for a CF partition. The number of CPs assigned to a CF partition should be kept to the minimum required to support the load.

A CF partition is most easily defined to LPAR/CE in terms of the LPAR host on which it is being run. For example, if the host is a 9021-962, then define the partition′s current processor as 9021-962.

Current thinking suggests that you should choose the LSPR MVS/ESA DB2 workload as the best LSPR SCP/workload alternative to represent a CF partition. MVS should be specified to LPAR/CE without implying AWM support; i.e., MVS/ESA (not MVS/ESA+). The LSPR workload should be specified as DB2.

To best represent the utilization of a CF partition, enter a value that is slightly less than LCP/N-way (number of LCPs defined to the CF partition divided by the N-way of the LPAR host). This calculation indicates to LPAR/CE that the partition will consume 100% of the cycles available to it. For example, in the case of the 9021-962, with one CP assigned to the CF partition, you would enter a utilization of 1/6 or 16.7%. This computed utilization must be factored down slightly to account for the intervening LPAR overhead. Use the information on Figure 104 on page 255 to determine the maximum utilization that will fit in this partition.

Note: Specifying a utilization for the CF partition that is low will not affect the Estimate of host LPAR-mode capacity / host B-mode capacity result. However, it will affect the Total capacity needed and Average utilization results.

If a CF partition's logical CPs are defined as shared (this is not recommended), its activity can have a significant impact the work being done in other shared partitions. Since the CF partition runs in "active wait," it will never receive less than its full relative share of CPU resource. This is true even when other partitions have a demand for the CPU resource and there is no coupling work pending for the CF partition. In this case, partition weights may need to be adjusted to assure the desired throughput. In LPAR/CE, you can adjust the partition weights to determine the CF partition's effect on other shared workload while viewing Figure 105 on page 255.

A.2.3.2 CFCC Using the ICMF Facility

CFCC using the ICMF facility emulates coupling links to allow communication between MVS/ESA SP-5.1 (and later) partitions and CF/I partitions running on the same physical CEC.

As originally delivered, CFCC in a CF/I partition also used an "active wait" process. Therefore, CF/I partitions had the same LPAR capacity planning considerations as do CF partitions (described above). In late 1995 a change was implemented which affects the performance of CFCC in a CF/I partition. This change implements a "non-active wait" process. The result of the enhancement is reduced system overhead when running a shared CF/I partition. (See WSC Flash 9528 for details.)

If this performance enhancement has been implemented, then you can take a more traditional view when using LPAR/CE to capacity plan when a CF/I partition is involved.

The partition's current CEC is most easily defined to LPAR/CE in terms of the LPAR host on which it is being run.

You should choose the LSPR MVS/ESA DB2 workload as the best LSPR SCP/workload alternative to represent the CF/I partition. MVS should be specified to LPAR/CE without implying AWM support; in other words, MVS/ESA (not MVS/ESA+). The LSPR workload should be specified as DB2.

Quantifying the utilization necessary to represent your CF/I partition activity will have to be based on user experience. Specifying a unrealistic utilization for the CF/I partition will not affect the Estimate of host LPAR-mode capacity / host B-mode capacity result. However, it will affect the Total capacity needed, and the Average utilization results. Using an unrealistic utilization will also make it difficult to accurately determine the effect of changing partition weights.

With the enhancement, a CF/I partition can be assigned a relative weight such that its weight per logical CP is on a par with the weight per logical CP of the MVS/ESA partition(s) communicating with it. There will be no unnecessary use of CPU cycles by the CF/I partition when there is no coupling work pending, thus ensuring more of the CPU resource remains available to the other partitions sharing the same CP resource. In LPAR/CE, you can adjust the partition weights to determine the CF/I partition's effect on other shared workload while viewing Figure 105 on page 255.

A.3 RMF Spreadsheet Reporter

RMF presents performance data in tabular form. One powerful tool for tabular data is the spreadsheet. The RMF Spreadsheet Reporter converts RMF Postprocessor output data to spreadsheet format, and provides a set of spreadsheet macros to graphically display the data. An example of an RMF Spreadsheet report is shown in Figure 107.

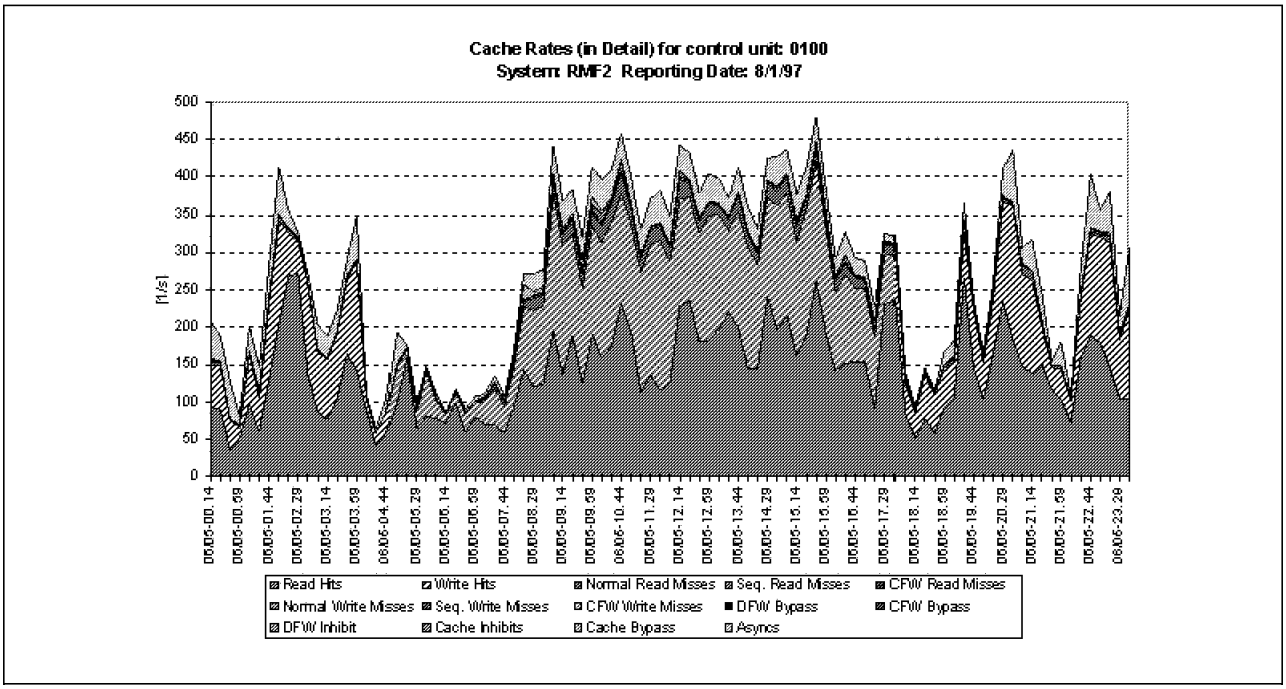


Figure 107. RMF Spreadsheet Reporter - Sample Cache Performance Overview in Spreadsheet

RMF Postprocessor Output

RMF measurement data collected in SMF records are typically used for reporting, historical performance analysis and capacity planning for OS/390 systems. The RMF Postprocessor which processes the data generates various types of reports.

Most commonly used are RMF reports for SMF data records type 70 to 78. In general, RMF reports are created for multiple RMF intervals or user-defined durations. The output is sent to the JES spool dataset and can be saved in a dataset (also referred to as the RMF listing).

RMF Overview Reports and Records

RMF Postprocessor reports are probably the most often used media to analyze RMF SMF data, but they are not the most efficient media. A more flexible way for historical reporting is provided with overview reports and records. Overview report and records are also generated by the RMF Postprocessor. The overview report is similar to the RMF Summary report, but you can define which data should be reported. Overview records contain the same information as the reports, but were especially designed to be used for further processing, for example by spreadsheet applications.

The Overview report and record is based on RMF Postprocessor exception control statements, thus providing a set of possible data items from which you can choose.

RMF Spreadsheet Converter

The RMF Spreadsheet Converter has been introduced with APAR OW06181 for RMF 4.3 and RMF 5.1. The Spreadsheet Converter converts RMF reports and RMF Overview records to a standard spreadsheet format. The standard format chosen is the Lotus 1-2-3 WK1 format which can be used by most spreadsheet applications. The spreadsheet converter is a workstation-based tool for the following workstation platforms:

- DOS
- Windows NT and Windows 95
- OS/2

RMF Spreadsheet Reporter

The RMF Spreadsheet Reporter is an extension to the RMF Spreadsheet Converter. The function helps you to process complete RMF listing data sets, as well as RMF Overview records. A wide variety of spreadsheet macros for Lotus 1-2-3 and Excel offers the capability to use common business applications to display and analyze RMF data (see Table 23).

Table 23. RMF Spreadsheet Reporter - Available Spreadsheet Macros

	Available Macros for		Available as	
Base RMF Report(s)	Lotus	Excel	part of OS/390 V2 R4 RMF	Add-on
RMF Summary report	√	√	√	
RMF CPU, Paging and Workload report (Compatibility Mode)	√	√	√	
RMF DASD report	√	√	√	
RMF Cache Subsystem report		√		√
RMF Coupling Facility report		√		√
RMF Overview record - System Summary	√	√	√	
RMF Overview record - Workload trend in compatibility mode	√	√	√	
RMF Overview record - Device trend	√	√	√	
RMF Overview record - Cache statistic trend (CUs and DASDs)		√		√

Note: The exception control statements necessary to generate the overview records are provided with the RMF Spreadsheet Reporter and can be generated from the Excel report macros which process the same resources.

The RMF Spreadsheet Reporter supports the same workstation platforms as the RMF Spreadsheet Converter. With OS/390 V2.4 RMF, the Windows version of RMF Spreadsheet Reporter is part of the RMF product. In addition, you can obtain the latest updates and the OS/2 version from the RMF home page on the Internet at the following address:

`http://www.s390.ibm.com/rmf`

IBM representatives may also request a copy from the RMFINFO tools disk:

- Version for Windows NT and Windows 95

 `TOOLS SENDTO BOEVM3 IIVTOOLS RMFINFO GET RMFPPNT PACKAGE`

- Version for OS/2

  ```
  TOOLS SENDTO BOEVM3 IIVTOOLS RMFINFO GET RMFPPOS2 PACKAGE
  ```

You can find a full description of the RMF Postprocessor and the RMF Spreadsheet Reporter in *OS/390 RMF User's Guide*, SC28-1949, and a description of the reports in *OS/390 RMF Report Analysis*, SC28-1950.

A.4 Performance Management Services (PMOS)

Performance management services are available to customers for OS390, CICS, DB2, and I/O subsystems. PM services provide a comprehensive analysis of subsystem workload behavior, utilization and responsiveness in the particular environment.

The PMOS service offering with respect to the Parallel Sysplex consists of the following parts:

PMMVS Performance analysis for a MVS system
PMIO Performance analysis for the I/O subsystem and batch windows
PMDB2 Performance analysis of a DB2 subsystem
PMCICS Performance analysis of a CICS subsystem

Additional service offerings are available for VM and SNA.

The PM service offerings use, as input, SMF data gathered by RMF and subsystems to provide reports which describe areas of contention or bottlenecks. Areas where latent demand exists are focused on in order to better understand what can be done to manage the system more effectively.

More information about PM services can be obtained from the Internet. The URL for the home page is:

`http://www.as.ibm.com/asus`

Or call:

`1-800-IBM-4YOU (1-800-426-4968)`

A.4.1.1 PMMVS - Performance Management MVS

PMMVS provides an analysis of workload activity, utilization and responsiveness in the OS/390 environment. The analysis is very much like a “health checkup.” The processor, processor storage and the I/O subsystem are analyzed to identify contention and bottlenecks. Areas where latent demand exists are focused on in order to better understand how they could be eliminated or managed more effectively. System overhead is identified, and options for elimination are discussed.

Service engagements can also optionally include processor sizing. Processor sizings project current workloads onto a target CEC using a composite ITR ratio. When projections are for 9672 CECs, additional reporting is done to address speed-of-the-engine issues for single-tasking address spaces.

PMMVS uses RMF and SMF data to report:

- Resource contention summary
- System activity summary
- Processor utilization
- Logical partition utilization, if applicable
- Central and expanded storage utilization
- Response time analysis for page/swap data sets
- Workload utilization of system resources
- Response Time analysis for TSO and batch workloads
- Input/output device utilization and response
- DASD cache controller performance
- CF utilization analysis, if applicable

- XCF signalling activity, if applicable
- Workload and resource utilization trends (optional)
- Processor sizing data (optional)

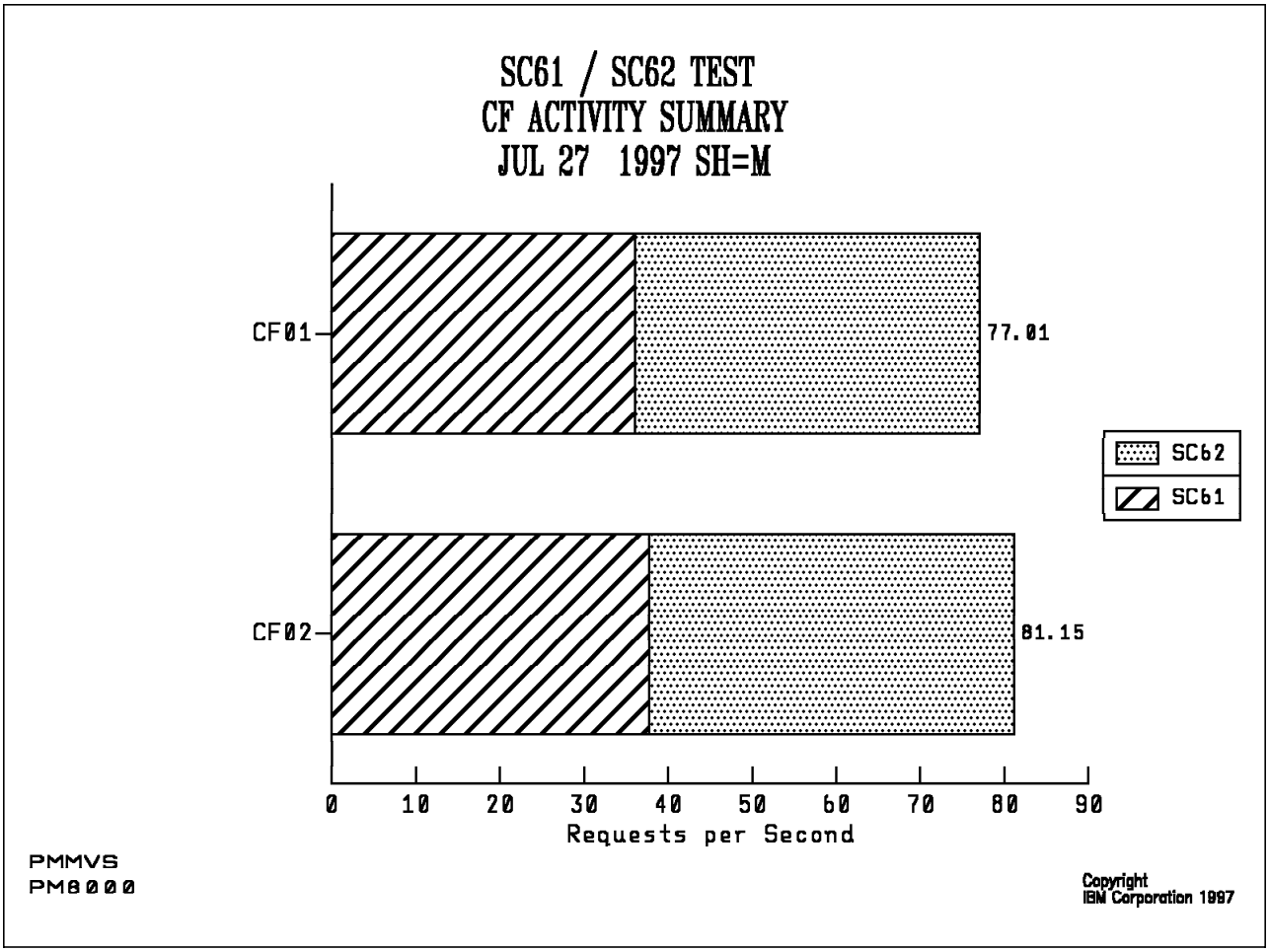


Figure 108. Average Requests by System for Each CF

Figure 108 illustrates the graphics contained in a PMMVS report. The PMMVS Analysis report comes as a booklet which contains detailed graphics and comprehensive descriptions for all analysis steps. Figure 108 shows the activity across the two CFs used in the sysplex.

A.4.1.2 Performance Management CICS (PMCICS)

PMCICS provides an analysis of subsystem workload activity, utilization and responsiveness in the CICS environment, focusing on subsystem and transaction usage of the processor, storage and I/O resources. This analysis helps you to better understand how well the applications are using existing resources, and to identify where contention and bottlenecks are occurring.

PMCICS uses RMF and CICS SMF data to report:

- Subsystem activity summary
- Subsystem response time summary
- Subsystem database activity summary
- Subsystem processor utilization
- Transaction activity analysis
- Transaction response time analysis

- Transaction wait time analysis
- Transaction database activity analysis

A.4.1.3 Performance Management DB2 (PMDB2)

A PMDB2 subsystem analysis focuses on resource usage at the system and the business application level. It examines a DB2 subsystem down to the data set level. This analysis helps you to better understand how well the applications are using existing resources, and to identify where contention and bottlenecks are occurring.

The analysis includes the following:

- DB2 subsystem resource contention and workload analysis
 - Overall workload usage of CPU, I/O and processor storage
 - Subsystem services efficiency including:
 - System event checkpoints
 - EDM and RID pool
 - DB2 Logging
 - Lock escalation
 - Virtual buffer pool and hiperpool
 - DB2 response time analysis for top plans, transactions and batch jobs
 - SQL profile for top plans, transactions and batch jobs
 - Data Sharing analysis
 - Global buffer pool
 - Global lock contention
 - Global locking wait time for top plans and transactions
- Optional Analysis, including DB2 system and user data set response time analysis

A.4.1.4 PMIO - Performance Management I/O

PMIO is aimed at the growing need for assistance with input/output (I/O) performance and batch tuning in the OS/390 environment. A PMIO analysis consists of an I/O analysis and/or a batch window reduction analysis. It can help you to determine which technology options can have the largest impact on improving your critical business systems.

I/O Subsystem Analysis: An I/O subsystem analysis focuses on your resource usage at the system and business application level. The service allows you to examine I/O subsystem tuning down to the data set level. The I/O view is sysplex-wide, meaning all systems sharing DASDs, SSIDs, and data sets are aggregated.

The analysis includes the following:

- CPU, processor storage, and I/O performance and usage
- VIO to expanded storage
- Hiperspace and VLF usage and benefits
- Control unit usage, contention cache controller, and volume analysis exploitation
- Data set performance analysis by I/O contention, storage classes, and business
- Applicable data in memory options
- Buffering and block size analysis (VSAM and non-VSAM)

- VSAM design
- BatchPipes/MVS usage
- Striping and compression candidates
- Sort analysis and exploitation

Batch Window Reduction Analysis: A batch window reduction analysis consists of a batch application suite analysis for one or more of your application suites and an optional batch job analysis for selected batch jobs. The batch application suite analysis helps you set the strategy and direction for batch window reduction by examining one or more selected application suites.

Your batch application suite analysis includes the following:

- A Gantt chart to illustrate the batch schedule and exploitation of parallelism (see the following chart)
- Job and step summary reports to highlight key components of job elapsed times
- Selection of specific candidate jobs for a batch job analysis

Job Name	Tape Mounts	23 Aug 1997 0•• 1•• 2••
B0		
B10SMF		
B11IO		
B12VSAM		
B12NVS		
B12HSM		
B12DSN		
BP2	7	
B2IOA		
B2IOB	2	
B2IOC	3	
B2MVSV		
B2PPTYP		
BP3		
B2DSNA		
BP4		
B2DSNB	4	
BP5		
BP6		

A batch job analysis includes the creation of job dossiers, with detailed information about each job analyzed. The batch dossier is sysplex-wide, meaning it contains data from all systems where the job ran.

This analysis includes the following:

- When the job ran
- The resources the job consumed
- The factor(s) causing the most job delays
- The method(s) the job used to access the data sets and any other job that accessed them
- The method(s) the job used DFSORT or Syncsort
- Performance from a DB2 perspective, analyzed by response time component

- Job initiator, allocation, and resource queuing delays
- The reliability of each job and system recover time
- Recommendations on improving each job including:
 - Buffering and block size changes
 - Use of VSAM Batch LSR
 - Use of Hiperbatch
 - Use of striping and/or compression
 - Tape mount management
 - Sort tuning
 - Job restructuring for increased parallelism
 - Use of BatchPipes/MVS

A.5 Enterprise Server Capacity Planning Oriented Tools Table

Table 24 on page 268 gives you an overview of some of the various capacity planning tools available in the S/390 environment.

To position CP2000, see Figure 109 on page 273. The table gives an overview of what aspects of capacity planning the tools can provide. The last column provides information on where to find more information in A.6, “Enterprise Server Capacity Planning Tools Catalogue” on page 271.

The table shows what areas of capacity planning the tools deal with. Distinction is made between the following topics:

- Parallel Sysplex
- CPU
- Processor storage
- DASD

If a tool provides information about a certain topic, this is shown with a checkmark (√). The column named “environment” gives information about what environment the tool will analyze.

Different tools require input from different sources and provide output in different ways as well.

For the various capacity planning tools, the table depicts the required input and the output available. For further information, look at the information in A.6, “Enterprise Server Capacity Planning Tools Catalogue” on page 271.

The column “Type” shows what type of tools we distinguish between:

Category	Definition
A	This tool is designed to analyze input data and to generate reports, graphs, and so forth as output.
CP	A tool which can predict system utilization and performance. The output varies from simple reports over single graphs to detailed documentations.
CPS	Capacity Planning Services is provided by IBM to support capacity and performance planning.
M	The primary purpose of this tool is to capture, process and store system data. For the most part, this tool also provides a facility to create reports or graphs.

Table 24 (Page 1 of 3). S/390 Capacity Planning Tools - Areas Covered

Name	Tool Type	Environment				Input	Output	Available from	Remarks	Reference
		Parallel Sysplex	CPU	Processor storage	DASD					
ARIAS (CPAIP)	A		√	√	√	RMF report, VMPF report, IPS	Reports, Graphs, EDF-file	EHONE		
BWAT	A	√	√			SMF records	Reports	MKTTOOLS		
BLSRAID	A		√	√	√	SMF records	Reports	ASKQ WSC		
CAA	A				√	GTF trace	Reports	MKTTOOLS		
CAU	A		√			GTF trace	Reports	Part of CICS V4.2		
CA17	CP		√	√	√	CICS CMF, IBM benchmarks	Reports	TXPPACKS at WINVMB		274
CBAT	CP	√	√			Manual, SMF, Reports	Reports	CAPRICE at HQVMIC1		274
CD13	CP	√	√			SMF records	Reports	TXPPACS at WINVMB		275
CP2000	CP	√	√	√	√	Manual, EDF-file	Reports, Graphs	EHONE		276
CP2000 Capacity Planning 2000	CP	√	√	√	√	Manual, EDF-file, CP2KEXTR output	Reports, Graphs	CPSTOOLS		276
PCR Processor Capacity Reference	CP		√			Manual	Graphs Tables	CPSTOOLS		277
CP2000 Quicksizer OS/2	CP	√	√	√	√	Manual	Reports	CPSTOOLS		278
CVRCAP CICS/VSAM RLS Capacity Planning	CP	√	√			SMF records	CP2000 Quicksizer input			278

Table 24 (Page 2 of 3). S/390 Capacity Planning Tools - Areas Covered

Name	Tool Type	Environment				Input	Output	Available from	Remarks	Reference
		Parallel Sysplex	CPU	Processor storage	DASD					
DB2PARA DB2 Parallel Sysplex Inputs for CP2000	CP	√	√			SMF records	DB2 update and delete activity			279
DCAT	CP				√	Manual, MVSXADAI output, VMDAI output	Reports	EHONE		279
DMAGIC/2	CP				√	Manual	Reports	MKTTOOLS		280
DSCAT	A				√		Reports	EHONE		
IMSAFAID IMS Affinity Aid	A	√				IMS commands and reports	Affinities			280
IMSSMQCP	A	√				IMS log tape	Report	MKTTOOLS		280
Performance Reporter for MVS	M/A	√	√	√	√	SMF records, Subsystem Logs	Reports, Graphs, DB2 tables	Program Product 5695-101	Replaces SLR	
HBAID	A		√	√		SMF records,	Reports	ASKQ, Local Support		
KGTV	CP	√	√			RMF report	Graph, table		Spread sheet	281
LPAR/CE	CP		√			Manual	Reports	CPSTOOLS		281
LSPR/PC	CP		√			Manual	Reports	MKTTOOLS		282
PMOS	CPS	√	√	√	√	SMF records	Reports, Graphs	PMOS		266
QCBTRACE	A	√	√	√			Reports, Graphs	MVSTOOLS		
RLSLKSZ	A	√				Access rates	VSAM CF structure size	MKTTOOLS		282

Table 24 (Page 3 of 3). S/390 Capacity Planning Tools - Areas Covered

Name	Tool Type	Environment				Input	Output	Available from	Remarks	Reference
		Parallel Sysplex	CPU	Processor storage	DASD					
RMF	M/A	√	√	√	√	SMF records, System Control Blocks	Reports, Graphs, SMF records	Program Product		
SAT	A			√		SMF/Extract file,	Reports, Graphs			
SLR	M/A	√	√	√	√	SMF records	Reports, Graphs,	Program Product		
SLR/CP	CP		√	√	√	SMF records	Reports, Graphs,	Feature of SLR		
SMF/Extract	M			√		SMF records	file	EHONE	Input for SAT	
SMFINFO	A		√	√	√	SMF records	Reports	ITSO Raleigh		
SMFVIEW	A		√		√	SMF records	Reports	ITSO Raleigh		
SMSSMA	A			√		Manual	Reports	EHONE		
SNAP/SHOT	CP/CPS	√	√	√	√	RMF report, IPS, ICS, SMF records, VM monitor	Reports			284
SOFTCAP	CP	√	√	√		Manual	Reports	MKTTOOLS		283
VMA	A	√				SMF	Reports	Part of DFSMS/MVS	Tape Mounts	284

A.6, “Enterprise Server Capacity Planning Tools Catalogue” on page 271 provides more detail about these capacity planning tools and services from IBM.

A.6 Enterprise Server Capacity Planning Tools Catalogue

This section gives you an overview of some IBM capacity planning tools, services, and program products that are valuable when doing capacity planning studies.

Many of the tools and services mentioned in this appendix are available for customer use. You can have capacity planning service performed for you by an IBM representative on a fee basis, as market support, or as a subcontract. Consult your IBM representative for more information about availability of the listed tools and services.

Note: This section does not provide a complete list of tools, as there are also IBM capacity planning tools available for many other areas including, but not limited to, networking, VM, VSE, RS/6000, AS/400, and Local Area Networks. Neither is this a comprehensive description of the unique tools.

This section contains information about:

- A.6.2, “Capacity Planning for CICS Cross Platform Migrations (CA17)” on page 274
- A.6.3, “CMOS Batch Analysis Tool (CBAT)” on page 274
- A.6.4, “CMOS Processor Utilization Tool (CD13)” on page 275
- A.6.5, “CP2000 (Capacity Planning 2000)” on page 276
- A.6.6, “Processor Capacity Reference (PCR)” on page 277
- A.6.7, “S/390 Parallel Sysplex Quick-Sizer (SPSSZR)” on page 278
- A.6.8, “CVRCAP (CICS/VSAM RLS Capacity Planning)” on page 278
- A.6.9, “DB2PARA (DB2 Parallel Sysplex Input for CP2000 Quick-Sizer)” on page 279
- A.6.10, “DASD Cache Analysis Tool (DCAT)” on page 279
- A.6.11, “DASD and Tape Magic/2 (DMAGIC2)” on page 280
- A.6.12, “IMSAFAID (IMS Affinity Aid)” on page 280
- A.6.13, “IMSSMQCP (IMS Shared Message Queue Capacity Planner)” on page 280
- A.6.14, “KGTV Tool, a CF Processing Model” on page 281
- A.6.15, “LPAR Capacity Estimator (LPAR/CE)” on page 281
- A.6.16, “Large Systems Performance Reference/PC (LSPR/PC)” on page 282
- A.6.17, “RLSLKSZ (RLS Lock Sizer)” on page 282
- A.6.18, “Capacity Effects of Software Migration (SOFTCAP)” on page 283
- A.6.19, “OS/390 UNIX Services Sizer Tool” on page 283
- A.6.20, “System Network Analysis Program/Simulation Host Overview Technique (SNAP/SHOT)” on page 284
- A.6.21, “Volume Mount Analyzer (VMA)” on page 284

A.6.1 Enterprise Server Capacity Planning and Other Tools Positioning

There are several tools and services available from IBM to aid the capacity planning studies. Several of these tools are mentioned in *Introduction to Storage Performance Management Tools and Techniques for MVS/ESA*, GG24-4045.

Capacity planning tools are available from several sources:

- TOOLS disks (MKTTOOLS, and so forth)

 For example, PCR, HBAID, BLSRAID, and BWAT. PCR is available on CPSTOOLS, and on IBMLink for approved Business Partners only.

- IBM services provided:

 For example, CPOS and so forth.

 These services are often characterized by being tailored to the individual situation. These services often use several of the tools mentioned above.

- IBM program products:

 For example, Performance Reporter for OS/390 which is a feature of Performance Data Manager for OS/390.

When using different tools for capacity planning, different amounts of time must be invested and different accuracy can be expected. Many factors have to be considered, among them are:

- Level of input-detail wanted or provided.
- Level of output-detail wanted or provided.
- Experience in use of a particular tool.
- Need for subsystem (for example DFSMS or CICS) modelling.
- Preciseness of forecast evaluation.
- Do I want to spend a lot of time?
- Do I want to spend a lot of money?
- Business importance or implication.

Figure 109 on page 273 illustrates the fact that all tools are not identical.

Note that other factors may influence your particular capacity planning instance; you should therefore look for more information about various tools before you make your choice.

Be aware that many of the tools mentioned are oriented toward performance management. They are usually not intended for stand-alone usage in a capacity planning study. However, they can play a key role in cooperation with CP2000, for instance.

Capacity planning and performance management are complementary processes within a single discipline. Many installations do not differentiate between the two. Often, much of the data needed for one or the other is used by both.

Capacity planning and performance management share many common tasks, even though the focus of each process is different. Capacity planning is concerned with what happens in the future. Performance management is more concerned with what is happening right now and why.

The Figure 109 on page 273 gives an overview for the area which is covered by each tool or method.

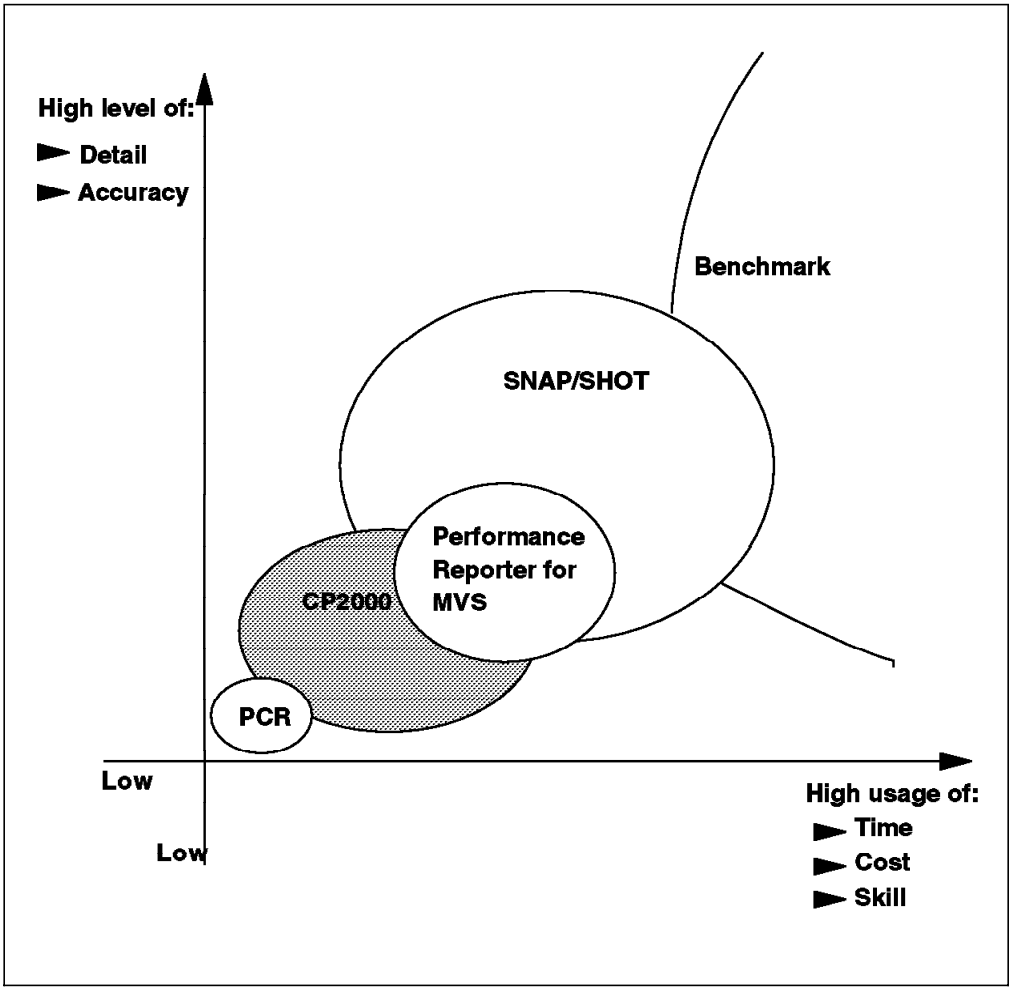


Figure 109. Positioning of Capacity Planning Tools. "Level of Detail" versus "Time Invested"

More detailed information on tools and on IBM program products and services in the capacity planning world is found in the following sections.

A.6.2 Capacity Planning for CICS Cross Platform Migrations (CA17)

NetReview and SNAP/SHOT Services in Raleigh have developed a methodology to assist customers with capacity planning for the migration of CICS for MVS/ESA applications to CICS/400, CICS/6000, or CICS/OS2.

In using this service, the client identifies which CICS transactions or groups of transactions are to be considered as migration candidates. Data reduction reports and/or simulation runs are then used to determine the throughput, capacity, and response time characteristics in one or more of the following environments:

- The CICS for AS/400, CICS for AIX or CICS for OS2 platform running the "migrated" CICS transactions.
- The CICS for MVS/ESA system running without the "migrated" transactions.
- The CICS for AS/400, CICS for AIX CICS for OS/2, and/or CICS for MVS/ESA platforms running modified versions of the transactions that incorporate client/server design alternatives.

The functional characteristics (that is, file control, temporary storage accesses) of existing CICS applications are captured from CICS for MVS/ESA, CICS/OS/VS, or CICS for VSE CMF performance data. Based upon the CMF and IBM benchmark data, SNAP/SHOT simulation models are developed for the desired platform (MVS, AS/400, SP2, or OS/2) to analyze various "what-if" scenarios.

CA17 package can be obtained from TXPPACS at WINVMB using the following command:

```
EXEC TOOLS SENDTO WINVMB TOOLS TXPPACS GET CA17 PACKAGE
```

A.6.3 CMOS Batch Analysis Tool (CBAT)

The CMOS Batch Analysis Tool (CBAT) is a WSC service that helps installations analyze their batch workloads as they prepare to migrate these workloads to the CMOS based 9672 CEC. This exciting service will provide capability to review batch workloads using:

- Information from any scheduler system, including OPC/ESA, CA, ZEKE and JES3. This information will be used to provide job interrelationships for the development of the schedule′s critical path and overall time line.
- Actual performance information of the scheduled batch jobs to allow estimations of how the jobs and job networks will run on the CMOS CECs.

The output of this service offering will be:

- Gantt chart of measured and estimated job net executions
- Pert charts showing critical path of measured and estimated job nets
- Resource histograms showing impacts on CP, tape drive allocations, and initiators
- A report with recommendations, and possible "what if" scenarios to adjust the job nets if either resources or time constraints are exceeded
- Consulting services to review and discuss the outputs of the CBAT service offering

For further information, send a note to:

```
IBMUSM24(CAPRICEJ)
```

A.6.4 CMOS Processor Utilization Tool (CD13)

The tool uses standard SMF data that is normally collected. Interval recording must be active, and the turning on of this option may be all the changes you need to make. If suitable interval recording is already in place, you will be able to use data that has been collected. The report should be reviewed to understand the workloads reported, and evaluate the likely impact. *It should not be assumed that because a workload has a high engine utilization that there is a problem.* The workload may use multi-tasking implicitly (exploit several TCBs), or you may be able to split the workload (for example splitting a large CICS system), or the workload may be of little importance and the increased run time may be acceptable. Each workload should be discussed and evaluated before moving the workload to the new CEC. CD13 can be obtained from TXPPACS at WINVMB using the following command:

```
EXEC TOOLS SENDTO WINVMB TOOLS TXPPACS GET CD13 PACKAGE
```

A.6.5 CP2000 (Capacity Planning 2000)

CP2000 is an OS/2 PC-based capacity planning tool to aid performance analysis and capacity planning for the S/390 environment. CP2000 is used to evaluate and project data processing capacity in the OS/390, MVS, VM, and VSE environments, supporting the ES/9000- and 9672-based CECs. CP2000 is used to do a capacity planning study on CECs and DASD. CP2000 will be updated to include all support in CP90 and more. At the time of writing, CP2000 support includes:

- SYSID CPU analysis
- SYSID DASD analysis
- Analyze processor speed interface
- "Interesting" DASD
- Workload analysis
- Parallel Sysplex (performance analysis only)
- Workload migration
- Simple growth specification
- Metrics Report (enhancements)
- LPAR analysis
- SYSID processor storage
- Enterprise DASD analysis
- Customize M-Value
- Merge workloads support
- Print power table (M-value table)
- Graphic support (enhancements)
- Workload support (enhancements)

IBM representatives can obtain the CP2000 overview, or CP2000 itself, from CPSTOOLS at WSCVM using the following commands:

```
OMNIDISK CPSTOOLS GET CP2KOVER PACKAGE

EXEC TOOLS SENDTO WSCVM TOOLS CPSTOOLS GET CP2KOVER PACKAGE

OMNIDISK CPSTOOLS GET CP2000 PACKAGE

EXEC TOOLS SENDTO WSCVM TOOLS CPSTOOLS GET CP2000 PACKAGE
```

For more information concerning CP2000, contact:

```
CP2000 at DALVM41B
Email:  cp2000@dalvm41.vnet.ibm.com
```

A.6.6 Processor Capacity Reference (PCR)

The Processor Capacity Reference (PCR) is a PC-based productivity tool intended to assess relative capacity for System 390 processors. Ratios are set with respect to any specific processor, and can be scaled in order to relate to other common capacity sources. Advanced functions are available to provide basic capacity planning information concerning workload growth, workload consolidation, and workload analysis. Results are presented in the form of graphs and tables, which can be used for presentations and handouts. PCR is based on IBM′s LSPR data provided by S/390 division.

PCR allows you to:

- Access to all IBM LSPR data
- Designate the base CPU and assign a scaling factor
- Select comparison processors for graphing
- Use filters to refine/limit processor lists
- Define a custom-mixed workload′s composition
- Add new processors to tables based on known relationships
- View capacity on a per-engine basis for N-way processors
- Analyze workload consolidation scenarios
- Generate graphic output
 - Relative capacity bar charts
 - Internal response time line charts
 - Mixed workload composition pie charts
- Graph customizing capability
- Create PM metafiles of graphs for use under FreeLance for OS/2
- Generate processor capacity table output for printing
- Save study scenarios
- Workload growth
 - Single growth rate for the entire workload over a given period
 - A system-wide default of 30% is supplied
 - Individual growth rates for the entire workload over the period
 - Individual growth rates for each of the defined business-units
 - Ability to define latent demand
- Assess workload characteristics and number of users supported

PCR runs under OS/2, and has been designed to take advantage of its graphical user interface. As a result, it is easier to use, and provides more function than LSPR/PC. A complete User′s Guide is included as part of the package. PCR should be used in conjunction with the LSPR document (SC28-1187), which provides background information about the capacity data being portrayed.

Note: Registration is required to use PCR (see the CPS rules, which are included with the package).

IBM representatives can obtain PCR from CPSTOOLS at WSCVM using either of the following commands:

```
OMNIDISK CPSTOOLS GET PCR PACKAGE

EXEC TOOLS SENDTO WSCVM TOOLS CPSTOOLS GET PCR PACKAGE
```

For more information concerning PCR, contact:

```
CP2000 at DALVM41B

Email:  cp2000@dalvm41.vnet.ibm.com
```

A.6.7 S/390 Parallel Sysplex Quick-Sizer (SPSSZR)

A CP2000 S/390 Parallel Sysplex Quick-Sizer tool is available, both in the host-based CP90, and as a PC-based OS/2 application. Minimal input is required, for instance describing current workload and the portion of the workload targeted for Parallel Sysplex implementation. Results provide a high-level estimate for the S/390 Parallel Enterprise Server configuration that would be required to support the designated workload, including the number of CECs, and storage. A high-level estimate for the number of CFs including the number of CPs, the number of coupling links, and storage size is given. The workload is analyzed for:

- Service time impact
- CEC response time
- Transaction response time

Note: This tool is also available as a host-based application on IBMLink/HONE/EHONE The IBMLink/HONE/EHONE fastpath command to get access to the Quick-Sizer tool is: CPSTOOLS

IBM representatives can obtain SPSSZR from CPSTOOLS at WSCVM using either of the following commands:

```
OMNIDISK CPSTOOLS GET SPSSZR PACKAGE

EXEC TOOLS SENDTO WSCVM TOOLS CPSTOOLS GET SPSSZR PACKAGE
```

For more information concerning SPSSZR, contact:

```
CP2000 at DALVM41B

E-mail:  cp2000@dalvm41.vnet.ibm.com
```

A.6.8 CVRCAP (CICS/VSAM RLS Capacity Planning)

CVRCAP executes under MVS and is a tool that helps with capacity planning for CICS/VSAM RLS applications. It simplifies the process of calculating the parameters that are needed as input to CP2000 Quick-Sizer. The data used by the tool is RMF type 70 and 72 SMF records and CICS type 110 SMF records.

This tool is available in some countries as an IBM service.

To obtain the CVRCAP description/package, authorized IBM representatives may execute either of the following commands:

```
OMNIDISK MKTTOOLS GET CVRCAP PACKAGE

TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET CVRCAP PACKAGE
```

A.6.9 DB2PARA (DB2 Parallel Sysplex Input for CP2000 Quick-Sizer)

DB2PARA is a tool that helps with capacity planning for DB2 data sharing applications. It simplifies the process of calculating the parameters that are needed as input to CP2000 Quick-Sizer. The data used by the tool is DB2 type 110 records.

The tool reports DB2 plan activity such as delete and update activity. Filtering and output tailoring is possible. Three scenarios are presented:

- Customized lock reduction based upon characteristics of the DB2 plans. Lock reduction of eighty percent obtained by plans with no update or delete activity.
- Generic, fifty percent lock reduction across all plans.
- No lock reduction, applicable if the input data represents a DB2 V4.1 system already employing type 2 indexes.

DB2PARA has no requirement for DB2PM.

This tool is available in some countries as an IBM service.

To obtain the DB2PARA description/package, authorized IBM representatives may execute either of the following commands:

```
OMNIDISK MKTTOOLS GET DB2PARA PACKAGE

TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET DB2PARA PACKAGE
```

A.6.10 DASD Cache Analysis Tool (DCAT)

The DASD Cache Analysis Tool (DCAT) is an application that IBM representatives may use to predict the performance of DASD subsystems with and without cache. DCAT predicts DASD response times and utilizations. From the entered configuration and performance information, DCAT builds an analytic model and uses this as a basis for performance predictions. Varying elements of the I/O subsystem produce reports that show how the performance of the subsystem would change.

The input has to be entered manually and comes from RMF summary reports, MVSXADAI, VMDAI, or CAA. The output provides results about performance information by volume or application.

To obtain the DCAT package, IBM representatives may execute either of following commands:

```
OMNIDISK MKTTOOLS GET DCATAIDS PACKAGE

EXEC TOOLS SENDTO MKTTOOLS MKTTOOLS TOOLS GET DCATAIDS PACKAGE
```

A.6.11 DASD and Tape Magic/2 (DMAGIC2)

DASD and Tape Magic/2 help you to estimate the performance of IBM DASD subsystems, or configure an IBM tape library. The DASD modeling capability within the DASD and Tape Magic/2 tool enables you to understand some of the performance and environmental differences between different IBM and non-IBM storage subsystems. It is an analytical performance modeling capability that uses performance information from an existing configuration. Together with information about subsystem characteristics, you can predict the likely performance of the same workload using a different storage subsystem.

The input from, for example the RMF DASD activity report, has to be entered manually and can be displayed in text form. You can also send or export the output data directly to Lotus 1-2-3/G or other graphic applications.

To obtain the DASD and Tape Magic/2 package, IBM representatives may enter the following command:

```
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET DMAGIC2 PACKAGE
```

A.6.12 IMSAFAID (IMS Affinity Aid)

IMSAFAID is a procedure that helps you to identify IMS affinities using existing IMS commands and reports. It discusses:

- How to use standard IMS commands to identify the DL/1 databases that are accessed by IMS transactions and BMPs to assist in IMS migration planning.
- PSB and transaction scheduling parameters, whose settings need to be checked to understand if the intent of these settings will be preserved in a parallel environment.
- How to identify IMS transactions that have an affinity to DB2.

To obtain the IMSAFAID package, IBM representatives may execute either of the following commands:

```
OMNIDISK MKTTOOLS GET IMSAFAID PACKAGE

TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET IMSAFAID PACKAGE
```

A.6.13 IMSSMQCP (IMS Shared Message Queue Capacity Planner)

IMS Shared Message Queue Capacity Planner is a tool that will help with the planning of CF capacity to support IMS shared message queues.

The only input required is the IMS log tape.

The tool produces a report that shows message rates that can be used as input to CP2000 Quick-Sizer.

This tool will provide a report of IMS message rates that can be input into CP2000 to estimate CF capacity needed to support IMS shared message queues.

To obtain the IMSSMQCP package, IBM representatives may execute either of the following commands:

```
OMNIDISK MKTTOOLS GET IMSSMQCP PACKAGE

TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET IMSSMQCP PACKAGE
```

A.6.14 KGTV Tool, a CF Processing Model

The KGTV Tool, A CF Processing Model, predicts the performance effect of the future changes in activity or configuration (or both) on the existing CF. Input to this tool are RMF reports. This tool is based on a net of queueing models M/M/c and G/G/c, and Allen-Cunneen approximation on G/G/c. KGTV tool is based on a spreadsheet.

KGTV is available from IIVTOOLS at BOEVM3. To obtain the KGTV package, IBM representatives may execute the following command:

```
EXEC TOOLS SENDTO BOEVM3 IIVTOOLS RMFINFO GET KGTV PACKAGE
```

For questions or additional information on KGTV, send a note to Robert Vaupel - IBMDE(VAUPEL)

A.6.15 LPAR Capacity Estimator (LPAR/CE)

The LPAR Capacity Estimator (LPAR/CE) is a PC-based tool designed to assess IBM CEC capacity when running various workloads in independent logical partitions under LPAR. This tool is based on IBM′s LSPR data and LPAR benchmark tests.

Input data required includes the planned PR/SM CEC model, and the following items defining each workload:

- Current CEC model
- SCP Version
- Workload type
- Utilization
- Proposed partition configuration under LPAR

Basic output represents the percent of the host LPAR CEC required to support the total workload, and a comparison of the CEC′s capacity, running in LPAR mode, to the same CEC′s capacity running in basic mode.

IBM representatives can obtain LPAR/CE from ″CPSTOOLS at WSCVM″ using either of the following commands:

```
OMNIDISK CPSTOOLS GET LPARCE PACKAGE
```

```
EXEC TOOLS SENDTO WSCVM TOOLS CPSTOOLS GET LPARCE PACKAGE
```

For more information concerning LPAR/CE, contact:

```
CP2000 at DALVM41B
```

```
E-mail:  cp2000@dalvm41.vnet.ibm.com
```

For questions or additional information on LPAR/CE, send a note to IBM′s Capacity Planning Services - DALVM41B(CP2000)

A.6.16 Large Systems Performance Reference/PC (LSPR/PC)

LSPR/PC is a PC-based productivity tool designed to provide relative capacity assessments for S/390 architecture CECs. The tool is based on IBM′s LSPR Internal Throughput Rate (ITR) data, and on the CEC comparison method described in *Large Systems Performance Reference (LSPR)*. Data includes:

OS/390, MVS/XA, and MVS/370 - E/S batch, Commercial batch, TSO, CICS, DB2, and IMS workloads

After choosing the specific CECs and workload of interest, LSPR/PC results are presented graphically, with supporting tables available. Graphs may be captured for use in presentations. Additional capabilities include the ability to scale the relative capacity data, project the life of a CEC with workload growth, estimate the number of users supported, and estimate the effects of a workload consolidation.

LSPR/PC is available to IBM representatives from PCTOOLS, or from MKTTOOLS. LSPR/PC is functionally stabilized but is still an official vehicle to inform the field about LSPR numbers change. LSPR/PC is running under DOS. A enhanced OS/2 Version of LSPR/PC CP2000 PCR can be found on CPSTOOLS at WSCVM. In opposition to LSPR/PC, CP2000 PCR is not functionally stable, for more information see A.6.6, “Processor Capacity Reference (PCR)” on page 277 and A.1, “Processor Capacity Reference (PCR) and Large Systems Performance Reference/PC (LSPR/PC)” on page 249. To obtain the LSPR/PC package, IBM representatives may execute either of the following commands:

```
OMNIDISK MKTTOOLS GET LSPRPC PACKAGE

TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LSPRPC PACKAGE
```

A.6.17 RLSLKSZ (RLS Lock Sizer)

RLSLKSZ estimates the CF storage requirements for the locking structure incurred in an CICS/VSAM RLS environment.
The input is:

- Number of CECs
- Capacity used on each CEC
- Read request fraction
- Sequential request fraction
- Recoverable transaction fraction
- Desired “false contention” fraction

The output is:

- Estimated LOCK structure size

This tools includes versions of this program for execution in OS2/DOS, VM/CMS, or AIX environments.

To obtain the RLSLKSZ package, IBM representatives may execute either of the following commands:

```
OMNIDISK MKTTOOLS GET RLSLKSZ PACKAGE

TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET RLSLKSZ PACKAGE
```

A.6.18 Capacity Effects of Software Migration (SOFTCAP)

This is an OS/2 REXX program that will show the effects on capacity when migrating to different software versions. Currently, CICS, DFP/DFSMS, IMS, OS/390, and VTAM are supported for a *non-data sharing* environment.

This tool was based on the work documented in WSC Flash 9441: *MVS/ESA V5.1 Performance Information V5 Release to Release Migration Software Performance Impact*.
To obtain the SOFTCAP package, IBM representatives may execute either of the following commands:

```
OMNIDISK MKTTOOLS GET SOFTCAP PACKAGE

TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET SOFTCAP PACKAGE
```

A.6.19 OS/390 UNIX Services Sizer Tool

The OS/390 UNIX Services Sizer tool which is currently tested is aimed to *size a S/390 system based on the capacity the application(s) need on a UNIX system*.
To run this estimate it is required that the IBM representative gets the following information:

- The vendor
- The number of UNIX systems
- Machine model and number of processors.
- The type of workload
- The utilization
- Percentage to be moved.

It might be that the application runs on a single UNIX system, or is distributed across several UNIX systems. For example,

- One runs the database server, the other runs the application server frontend.

 In this case you have to provide the information listed above for each of the two UNIX systems.

- Both, the database server and the application front end run on one UNIX system.

 In this case you have to figure out how the utilization is distributed between the two parts.

Your IBM representative can contact one of the specialist who has the OS/390 UNIX Services Sizer tool available. A list is kept on the intranet at

```
http://w3.s390.ibm.com/s390ncc/nccinfo
```

A.6.20 System Network Analysis Program/Simulation Host Overview Technique (SNAP/SHOT)

SNAP/SHOT is a capacity planning tool and service used to simulate the operation of a real system. It reports end-to-end response time, capacities, bottlenecks in host, queues, network, and I/O environments. This tool is essential in explaining the behavior of new or existing workloads.

SNAP/SHOT can estimate transaction response times, when you are pertaining data sharing in a Parallel Sysplex. SNAP/SHOT services provide Parallel Sysplex simulation, including CF and CICSplex. Processing resources related to MRO, ISC, CICS dynamic transaction routing, CICS mirror transactions, and XCF are other elements that can be simulated.

SNAP/SHOT supports multiple environments and products:

- OS/390, OS/400, AIX/6000, OS/2
- ES/9000, ES/9072, AS/400, RS/6000, SP2, S/88, PS/2, selected OEM CECs
- CICS, IMS, DB/2, TSO, selected OEM DBMS
- ETHERNET, Token-Ring
- SNA, TCP/IP, APPN, VSAT, X.25, FRAME RELAY, NetBIOS
- Hubs, routers, gateways, bridges
- Client applications

For an overview of this service and information about who to contact, have your IBM representative send a note to:

HQVMIC1(SNAPSHOT)

A.6.21 Volume Mount Analyzer (VMA)

The volume mount analyzer is a tool that can help you analyze your installation's current tape environment.

The volume mount analyzer produces reports that profile your tape mount workload and tape media usage, so that you can decide whether tape mount management might benefit your installation. In addition, the volume mount analyzer identifies data sets that, because of their large size, can benefit from advanced cartridge hardware and media technology. The volume mount analyzer uses your installation's SMF data to analyze tape mount activity and to produce reports that help you:

- Identify trends and other information about tape mount events, including data set name, job, program, and data set size (bytes transferred).
- Evaluate the tape hardware configuration.
- Quantify the benefits of tape mount management in terms of library and tape mount reduction.
- Determine which data sets are good candidates for tape mount management.
- Determine data class and management class requirements for tape mount management candidates
- Develop ACS routine filters to select tape mount management candidate and exclude other data sets that must remain on tape.
- Determine the size of the DASD buffer and the high and low thresholds needed for the buffer's storage group.

VMA is available as part of DFSMS/MVS.

Appendix B. Processor Storage Management and Extended Storage

If DASD paging is negligible, do not define ES unless you use hiperbatch or you can live with DFSORT performance. If DASD paging is significant, do define ES by splitting the storage at UIC value of 254. The rationale for this theory is explained in the following sections.

In 9672 systems you may face situations where your next step to increase storage capacity is big, for example from 1 GB to 2 GB. This step may or may not always be economically possible. If you experience *considerable paging*, then you are advised to look closer at the following CS/ES split cases.

As seen earlier, there are few reasons to define expanded storage. But there are cases when redefining storage from central to expanded storage may *decrease the paging to DASD.* Figure 110 describes the scenario.

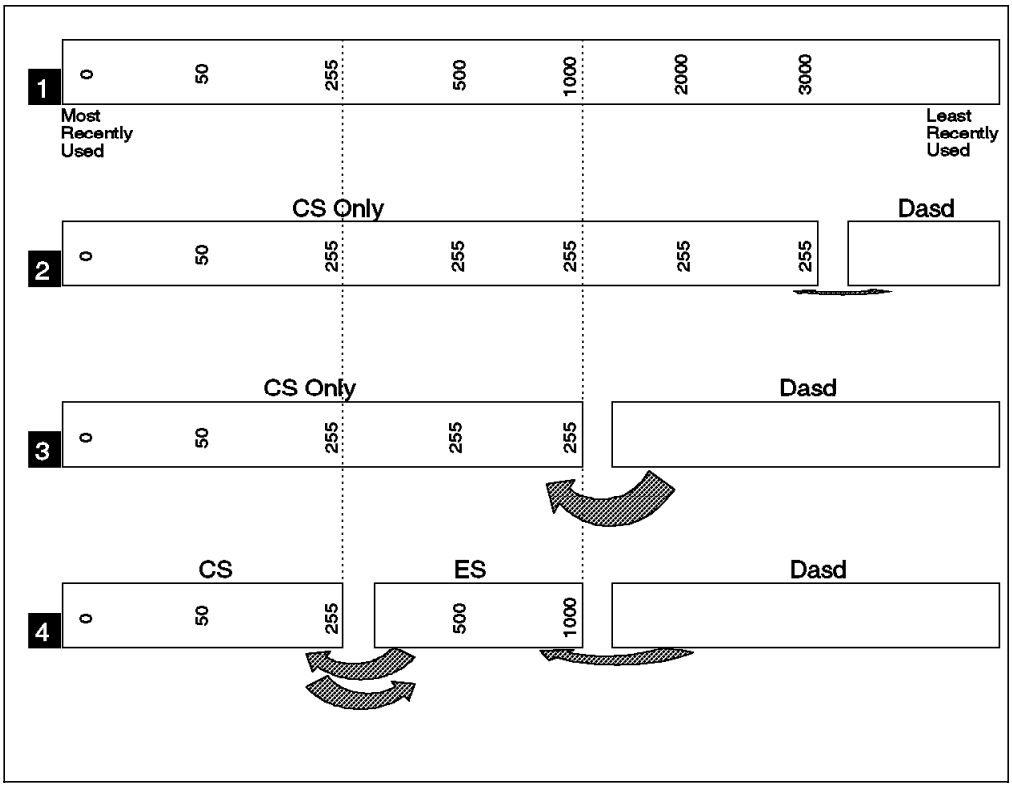


Figure 110. Storage Pages and CS/ES Storage

In the first row (1), all the storage pages of a system are presented and sorted according to their age. On the left are *most recently used* pages. These pages have a low Unreferenced Interval Count (UIC) value. On the right are the *least recently used* pages. These pages have a high Migration Age (MA) value.

In the second row (2) is the “ideal” scenario, when you have “enough” storage in your system. Here the DASD page fault rate is very small, say less than a few pages per second. In this scenario there is *no need to define expanded storage,* unless there are *special requirements* for it.

In the third row (3) is a case when the system has “too little” storage; this causes the DASD paging to be high; the page fault rate is maybe hundreds of

© Copyright IBM Corp. 1998

pages or more per second. In this scenario *no expanded storage is defined.* All pages with a “high” age have a UIC value of exactly 255. The system treats all pages having a UIC value of 255 in the same manner. This means that the Least Recently Used (LRU) algorithm does not work “optimal.” Total paging to DASD may *increase* somewhat due to this effect.

In the fourth row (**4**) is a scenario when the system again has “too little” storage. In this case you have defined expanded storage. Now the LRU algorithm in CS is working properly and it covers the pages with the age from 0 to 255. Equally the LRU method works in ES as well, and it covers the pages with the ages from 255 to 1000 here. This means that the system can “distinguish” between pages having an age over 255. The least used pages are paged out and the DASD paging is kept to a minimum. One side effect of the page movement between CS and ES storages is CPU overhead. If the CS-ES page movement rate is low, or the CPU utilization is lower than, say, 90% this is not significant.

The split between CS and ES storages should ideally be done at a point where the UIC value equals 254, or slightly less than 255.

Note: It is of course not possible to always split exactly at this point in your system due to workload fluctuations and so on. However, this discussion provides some ideas on how to size ES and CS given the cases illustrated above.

Also observe that this method probably works better for TSO and Engineering/Scientific (S/E) oriented workloads rather than On-Line Transaction Processing (OLTP) environments using large amounts of in-storage buffers

This is an iterative process where you start out by defining a certain amount of expanded storage, monitor the results, then redefine your storage definitions until the desired result appears.

Appendix C. Providing Input to CP2000

This Appendix gives an introduction to the various input mechanisms to CP2000.

There are different ways to provide input to CP2000, for example:

1. Manual input

 It is possible to manually enter all the data necessary for input for CP2000 from the RMF reports. This means a lot of data entry, with many possibilities for errors. This method of providing input is not recommended.

 Manual input is not discussed further here.

2. Using CP2KEXTR

 Note: The CP2KEXTR program is the recommended method of getting data into CP2000. Other methods include PMMVS and MXG-Bridge.

 CP2KEXTR will create EDFI-files, which can be processed by CP2000. CP2KEXTR reads SMF7x records (RMF records). CP2000 optionally processes SMF30 and SMF42 and IPS parameters. CP2KEXTR reads SMF type 42.6 and type 245 records. This provides a means of extracting CRR data for the IBM DMAGIC2 tool.

 A description on how to provide input to CP2KEXTR is described briefly here.

The following items are shortly discussed here:

- SMF, which actually writes the SMF7x records
- RMF Monitor I, which creates the SMF7x records
- RMF Postprocessor, which writes the RMF reports
- CP2KEXTR, how to get it, and how to provide input to CP2KEXTR
- The IPS and ICS parameters for grouping the workload
- Workload manager definitions, and workloads and service classes for grouping the workload
- Other information, like DASD configuration, for additional Capacity Planning information

C.1 SMF Parameters

It is necessary to check the SMF parameters. The following is a sample set of parameters for a member SMFPRM01:

```
ACTIVE
DSNAME(SYS1.MAN1,SYS1.MAN2,SYS1.MAN3)
NOPROMPT
REC(PERM)
MAXDORM(3000)
STATUS(SMF)                                  <=== 2
JWT(0030)
SID(SYSA)
LISTDSN
LASTDS(MSG)
NOBUFFS(MSG)
SYS(TYPE(0:69,70:79,80:245)                  <=== 1
    EXITS(IEFU83,IEFU84,IEFACTRT,IEFUJV,IEFUSI,IEFUJI,IEFUTL,IEFU29),
    INTERVAL(SMF(SYNC)),NODETAIL)            <=== 2
SUBSYS(STC,EXITS(IEFU29,IEFU83,IEFU84,IEFUJP,IEFUSO,
    IEFUJI,IEFACTRT))
SUBSYS(JES2,EXITS(IEFUJI,IEFACTRT,IEFU83))
INTVAL(30)                                   <=== 2
SYNCVAL(0)                                   <=== 2
```

1 Record types 70-79 should be included here.

© Copyright IBM Corp. 1998

2 Starting from MVS 4.3, it is possible to synchronize SMF recording and the other recorders globally according to SMF. This is highly recommended in general and mandatory in the Parallel Sysplex.

C.2 RMF Monitor I

RMF Monitor I writes SMF7x records (RMF records) to SMF data sets, as required. This recording is controlled by RMF parameters in SYS1.PARMLIB member ERBRMFxx.

Note that RMF Monitor III also writes SMF7x records:

XCF (74.2)
OS/390 UNIX Services (74.3)
CF (74.4)

The following is a sample of ERBRMF00:

```
CPU
CHAN
DEVICE(DASD)
IOQ
INTERVAL(30M)                          <=== 1
ENQ(D)
NOOPTIONS
PAGESP
PAGING
RECORD                                 <=== 2
NOSTOP
SYNC(SMF)                              <=== 3
VSTOR
WKLD(PERIOD)                           <=== 4
```

1 Recording interval should be 60 minutes or less.
2 Recording to SMF data sets is necessary.
3 Synchronizing according to SMF is possible starting from MVS V4.3 This is recommended. If your system is at lower level, you can instead specify:

```
SYNC(OM).
```

4 WKLD parameter is necessary.

C.3 RMF Postprocessor

RMF Postprocessor writes the RMF reports to DASD for later processing.

The following is sample JCL used to copy the RMF records from a weekly SMF data set using IFASMFDP, and to write the reports to paper and to DASD using RMF Postprocessor:

```
//USERX JOB (11111,USER),'             ',NOTIFY=USER,
//   REGION=4M,CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1)
//SMFDP    EXEC PGM=IFASMFDP
//INDD1    DD  DSN=SYSA.SMFDUMP.WEEK,DISP=SHR
//OUTDD1   DD  DSN=USER.RMFA.DATA,DISP=OLD
//*** DCB=SYSA.SMFDUMP.WEEK,SPACE=(CYL,(5,5)),UNIT=DASD
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
SID(SYSA)
DATE(96192,96196)
START(0000)
END(2359)
INDD(INDD1,OPTIONS(DUMP))
OUTDD(OUTDD1,TYPE(70:79,254))
/*
//POST1   EXEC PGM=ERBRMFPP,REGION=1000K
//MFPINPUT DD  DSN=USER.RMFA.DATA,DISP=SHR
//MFPMSGDS DD  SYSOUT=*
//*
//*  THE FOLLOWING IS FOR A 5 DAYS REPORT ON PAPER
```

```
//*
//SYSIN    DD  *
DATE(96192,96196)
STOD(0700,1759)
SUMMARY(INT)
RTOD(1300,1359)                                <=== 1
DINTV(0500)                                    <=== 1
REPORTS(CPU)
REPORTS(PAGING)
REPORTS(PAGESP)
REPORTS(IOQ)
REPORTS(DEVICE(DASD))
REPORTS(WKLD(GROUP,SYSTEM))
SYSOUT(A)
/*
//POST2   EXEC PGM=ERBRMFPP,REGION=1000K
//MFPINPUT DD  DSN=USER.RMFA.DATA,DISP=SHR
//MFPMSGDS DD  SYSOUT=*
//*
//*  REPORTS TO DASD
//*
//MFR00001 DD  DISP=OLD,DSN=USER.RMFDD1.LST, <=== 2
// DCB=(RECFM=VBA,LRECL=137,BLKSIZE=6028),
// SPACE=(TRK,(5,5)),UNIT=DISK
//MFR00002 DD  DSN=USER.RMFDD2.LST,DISP=OLD
//MFR00003 DD  DSN=USER.RMFDD3.LST,DISP=OLD
//MFR00004 DD  DSN=USER.RMFDD4.LST,DISP=OLD
//MFR00005 DD  DSN=USER.RMFDD5.LST,DISP=OLD
//PPSUM001 DD  DSN=USER.RMFSUM.LST,DISP=OLD
//SYSIN    DD  *
DATE(96192,96196)                              <=== 3
STOD(0700,1759)                                <=== 4
SUMMARY(INT)
RTOD(1300,1359)                                <=== 5
DINTV(0100)                                    <=== 5
REPORTS(CPU)
REPORTS(PAGING)
REPORTS(PAGESP)
REPORTS(IOQ)
REPORTS(DEVICE(DASD))
REPORTS(WKLD(GROUP,SYSTEM))
SYSOUT(A)
//
```

1 The paper copy is a summary of all the periods from 13:00 to 14:00 hours.
2 These reports will be printed to DASD. In this example each unique report is written to its own sequential data set.

Another possibility is to write the reports to a partitioned data set (PDS), with each report acting as an individual member of this PDS. But then, write the summary report to a separate data set, or on a separate step to the same PDS, to avoid locking problems.

A third possibility is to write the reports to the same sequential data set one after the other, specifying DISP=MOD parameter.

3 Measurements are for five consecutive days, covering one week.
4 The summary report is produced for the prime shift.
5 In this case, the hour from 13:00 to 14:00 is selected, the peak hour that is the base for this capacity plan.

For sample JCL to obtain

- Goal mode reports
- Coupling Facility Activity reports
- XCF activity reports

refer to *OS/390 RMF User's Guide*.

C.4 RMF Monitor III

Remember to use RMF Monitor III, as it is very powerful in monitoring and reporting system activity and system status. This will give you a good knowledge about the system.

Note: The data gatherer of RMF Monitor III is required to get the RMF Coupling Facility reports based on SMF Type 74 records.

C.5 CP2KEXTR

CP2KEXTR is a program which run on an MVS system to build a file for auto input in CP2000. You may run this program, download the file to diskette, then upload it to a PC.

CP2KEXTR directly reads SMF7x records (RMF records) and produces the EDFI-file, which is the input for CP2000 processing. CP2KEXTR is able to use IPS parameters as well.

Refer to *CPSTOOLS disk* to obtain documentation on CP2KEXTR program and for getting the program itself.

The following is not a description of CP2KEXTR, but it is sample JCL that selects RMF records from a weekly saved SMF data set to a data set for CP2KEXTR processing:

```
//USERX JOB (11111,USER),'              ',NOTIFY=USER,
//   REGION=4M,CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1)
//SMFDP    EXEC PGM=IFASMFDP
//INDD1    DD  DSN=SYSA.SMFDUMP.WEEK,DISP=SHR
//OUTDD1   DD  DSN=USER.RMFA.DATA,DISP=OLD
//*** DCB=SYSA.SMFDUMP.WEEK,SPACE=(CYL,(5,5)),UNIT=DASD
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
SID(SYSA)
DATE(96192,96196)
START(0000)
END(2359)
INDD(INDD1,OPTIONS(DUMP))
OUTDD(OUTDD1,TYPE(30(2,3),42(6),70:75,78,254)
```

You may run the Extract program against the original SMF file if you still have it. In this case, to keep this data, set up the Extract parameter to SAVE=YES or SAVE=ALL. The next piece of sample JCL shows you to filter the input EDFI data, for processing with CP2KEXTR.

```
//USERX JOB (11111,USER),'              ',NOTIFY=USER,
//   REGION=6M,CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1)
//EXTR EXEC  PGM=CP2KEXTR                                          0003
//STEPLIB DD DSN=???????.????.????,DISP=SHR                        0003
//SYSPRINT DD SYSOUT=*                                             0003
//SMFIN  DD DSN=PAGEB.RMF.DATA,DISP=OLD                            0003
//EDFFILE DD DSN=?????.????.EDFI(PROD),DISP=OLD                    0003
//BCUMAP DD DSN=?????.CP2KEXTR.DATA(PRODBCU),DISP=SHR              0003
//PGNMAP DD DSN=?????.CP2KEXTR.DATA(PRODPGN),DISP=SHR              0003
//MAGIC  DD DSN=?????.MAGIC.DATA(PROD),DISP=OLD                    0003
//SYSIN DD *                                                       0003
ENT='CP2000 INTERNATIONAL INC.'                                    0003
CECID=LPAR
SYSID=PROD
DURATION=1
TIME=(09-17)                                                       0003
DATE=(10/10/96-10/14/96)                                           0003
WORKLOAD=YES
BCU=YES
TYPE42=YES
/*                                                                 0003
```

C.6 ICS/IPS Parameters

CP2KEXTR reads these parameters directly from the parameter library.

Both IPS and ICS parameters should be printed out for the capacity planner.

C.7 Configuration Data

Remember to collect the information about CPU, main storage and DASD. Most of the information exists on RMF reports, for example some information such as DASD Subsystem Cache size is not printed there.

C.8 Knowledge of Workloads

You have to have a basic understanding of the involved workloads. This will enable a more precise analysis, and will add more weight to your judgements.

Appendix D. M-value Table (December 1997)

This appendix gives the current M-value for each IBM CEC for the OS/390 operating system. This table is current as of September 1997.

The script source for this table can be generated from CP2000:

1. Access CP2000 on your PC, click on `CP2000 Full Function`
2. Click on `File` menu bar on CP2000 Processor Inventory window
3. Click on `Print Power Table`
4. Specify .HTM file to contain M-value table
5. Click `OK`
6. Use a browser (for example, Netscape or WordPro) to display and print the M-value table in the .HTM file.

Table 25 (Page 1 of 3). M-values (December 1997)

CEC Model	Current M-value	Number of CPs	CPU Model	Current M-value	Number of CPs	CPU Model	Current M-value	Number of CPs
Generation 4 CMOS								
9672-RY5	19416	10	9672-RX5	17309	10	9672-R95	16521	9
9672-R85	15567	8	9672-R75	14384	7	9672-R65	13001	6
9672-R55	11439	5	9672-R45	8997	4	9672-R35	7100	3
9672-RC5	6289	3	9672-R25	5033	2	9672-RB5	3818	2
9672-R15	2710	1	9672-RA5	2129	1			
Generation 3 CMOS								
9672-RY4	16054	10	9672-RX4	14595	10	9672-R94	13854	9
9672-R84	12907	8	9672-R74	11780	7	9672-R64	10489	6
9672-R54	9048	5	9672-R44	7474	4	9672-R34	5798	3
9672-RC4	2096	1	9672-R24	1448	1	9672-RB4	2741	2
9672-R14	5185	3	9672-RA4	4010	2			
Multiprise 2003-2xx								
2003-257	6756	5	2003-247	5931	4	2003-246	5310	4
2003-237	4665	3	2003-2C5	3805	3	2003-227	3340	2
2003-225	2830	2	2003-224	2178	2	2003-116	3626	1
2003-216	1623	1	2003-215	1304	1	2003-207	1078	1
2003-206	714	1	2003-205	555	1	2003-204	395	1
2003-203	237	1						
Multiprise 2003-1xx								
2003-156	6064	5	2003-146	5314	4	2003-136	4182	3
2003-1C5	3821	3	2003-135	3344	3	2003-126	3002	2
2003-125	2394	2	2003-124	2219	2	2003-116	1626	1
2003-115	1323	1	2003-107	1078	1	2003-106	714	1
2003-105	555	1	2003-104	395	1	2003-103	237	1
2003-102	158	1						
9672 R2/R3								

© Copyright IBM Corp. 1998

Table 25 (Page 2 of 3). M-values (December 1997)

CEC Model	Current M-value	Number of CPs	CPU Model	Current M-value	Number of CPs	CPU Model	Current M-value	Number of CPs
9672-RX3	7442	10	9672-R83	6425	8	9672-R73	5820	7
9672-R63	5153	6	9672-R53	4398	5	9672-R72	4703	7
9672-R52	3773	5	9672-R42	3177	4	9672-R32	2499	3
9672-R22	1750	2	9672-R12	936	1	9672-RA2	658	1
9672 R1								
9672-R61	2811	6	9672-R51	2509	5	9672-R41	2144	4
9672-R31	1710	3	9672-R21	1213	2	9672-R11	658	1
9021								
9021-9X2	20807	10	9021-982	17417	8	9021-972	15651	7
9021-962	13801	6	9021p982	11327	5	9021-952	11810	5
9021-942	9625	4						
9021-941	9267	4	9021-832	7397	3	9021-831	7177	3
9021-822	5037	2	9021-821	4943	2	9021-711	2698	1
9021-900	10526	6	9021-860	9046	5	9021-820	7383	4
9021-822	5087	2	9021-821	4966	2			
9021-740	5406	3	9021-720	5222	6	9021-660	3912	2
9021-711	2696	1						
9021-640	3774	2	9021-620	3649	4	9021-580	2857	3
9021-520	2058	1	9021-500	1960	2	9021-340	1047	1
9021-330	946	1						
9121								
9121-742	4683	4	9121-732	3661	3	9121-190	314	1
9121-180	214	1	9121-622	2541	2	9121-621	2542	2
9121-522	2032	2	9121-521	2065	2	9121-511	1362	1
9121-411	1078	1	9121-311	886	1	9121-610	3010	4
9121-570	2338	3	9121-490	1624	2	9121-480	1619	2
9121-440	1255	2	9121-320	881	1	9121-260	672	1
9121-210	465	1						
9221								
9221-421	1260	2	9221-221	780	2	9221-211	700	1
9221-201	580	1	9221-191	468	1	9221-200	441	2
9221-170	264	1	9221-150	202	1	9221-130	139	1
9221-120	78	1						
3090								
3090-28T	1920	2	3090-25T	1201	2	3090-17T	784	1
3090-15T	664	1	3090-18T	1047	1	3090-600J	5222	6
3090-600S	4406	6	3090-600E	3262	6	3090-500J	4456	5
3090-500S	3794	5	3090-500E	2946	5	3090-400J	3649	4
3090-400S	3134	4	3090-400E	2442	4	3090-380J	2796	3

Table 25 (Page 3 of 3). M-values (December 1997)

CEC Model	Current M-value	Number of CPs	CPU Model	Current M-value	Number of CPs	CPU Model	Current M-value	Number of CPs
3090-380S	2428	3	3090-300J	2857	3	3090-300S	2539	3
3090-300E	1974	3	3090-280J	1920	2	3090-280S	1681	2
3090-280E	1317	2	3090-250J	1038	2	3090-250S	952	2
3090-200J	1960	2	3090-200S	1750	2	3090-200E	1372	2
3090-180J	1047	1	3090-180S	946	1	3090-180E	744	1
3090-170J	744	1	3090-170S	659	1	3090-150J	582	1
3090-150S	528	1	3090-150E	457	1	3090-120J	404	1
3090-120S	339	1	3090-120E	339	1	3090-110J	339	1
3090-100S	230	1						
4381								
4381-92E	379	2	4381-91E	201	1	4381-90E	160	1

Appendix E. Queueing Theory Concepts

At a Glance

This chapter presents elementary information about queueing systems. Everyone knows that as systems become busier queuing time increases very quickly. In this section we show that in a single server system, if the server is observed at a random time to be busy, the average time for service to complete (the Mean Residual Wait Time) is always more than half the average service time unless the service time is constant.

We apply this result to see that the service time for synchronous CF requests can be significantly increased by delays due to asynchronous CF requests.

Recommended Reading:

- *S/390 MVS Parallel Sysplex Performance*, SG24-4356
- *Arnold O. Allen: Probability, Statistics, and Queueing Theory with Computer Science Applications*

© Copyright IBM Corp. 1998

E.1 A Taxi Journey Example

Suppose a taxi always makes one journey each per hour of duration 1,2,3,4 and 10 minutes. The average length of a journey is (1+2+3+4+10)/5 = 4 minutes. Suppose at some point we notice the taxi is busy, how long on average will it be until the taxi is free?

The most obvious possible solution is that it is half the average journey time, in other words 2 minutes. However, as we will see, this is incorrect.

If we imagine that the journeys were of 1,2,3,4 seconds and 10 minutes then the average journey time becomes 2 min 2 seconds and half the average journey time is 1 min 1 sec, but you probably suspect that you are more likely to be waiting on a 10 minute journey than one of 1 second. We realize our mistake. We did not take into account that the probability that the taxi is on a long journey is greater than the probability that it is on a short journey. Returning to our original assumptions the probability that the taxi is busy on, say, a 3 minute journey is 3/(1+2+3+4+10) and if it is, then we expect to wait on average 3/2 minutes. Similar expressions hold for the other durations. In general, if we let t_j denote a typical time we can express the time we expect to wait if the taxi is on a journey of length t_j is

$$\frac{t_j \times \frac{t_j}{2}}{\sum_{i=1}^{5} t_i}$$

Therefore the (weighted) average value of the weight times is

$$\frac{\sum_{i=1}^{5} t_j \times \frac{t_j}{2}}{\sum_{i=1}^{5} t_i}$$

If we divide the top and bottom of this expression by 5 we find
RMWT=E(T^2)/(2 x E(T)) (R)
where RMWT is the Residual Mean Wait Time and T denotes a random variable with the distribution above, and E is the expectation operator (ie E(T) is the average of the journey times and E(T^2) is the average of the squares of the journey times). But in fact we have not used anything particular to our distribution and this formula is correct in general. It can be expressed by saying the RMWT is the mean square journey time divided by twice the average journey time.

For our distribution, we find E(T^2) is 130/5=26 and E(T) (the average) is 4 and so the RMWT is 1/2 x 26/4=3.25 minutes.

If we change the numbers so that the average stays the same, we can increase the RMWT. For example if the journeys are 0.5,1,1.5,2 and 15 minutes, then the mean square is (225+7.5)/5=46.5, the average as before is 4 minutes and so the RMWT is 46.5/(2x4)=5.8125.

You may be familiar with this effect in a bank with one teller, the customer he is serving always seems to be doing someyhing complicated! It also helps explain why when you ring an engaged phone number they never seem to be having a short call.

Allen in the book mentioned above, discusses a different formulation of this 'Taxi Paradox'. and quotes (R) without proof, (in the form (RR) below).

E.2 Relation between Mean Square, Variance and Coefficient of Variation

It is a simple result in statistics that if X is any random variable then
$Var(X)=E(X^2)-E(X)^2$ (*)

The Coefficient of Variation of X, C_X is defined as $C_X=Var(X)/E(X)^2$

Rearranging (*) and dividing throughout by $E(X)^2$ we find
$E(X^2)/E(X)^2= 1 + C_X$

Using our definition of RMWT and equation (R), we now find
$RMWT=E(X)*(1+C_X)/2$ (RR)

Now we see why we normally have to wait for more than half the average server time when we find the server (such as a phone or a CF link), busy. The wait time remaining is half the average service time only if $C_X=0$. This happens only if VAR(X)=0, that is the service time is constant.

For a Markovian distribution, (the usual sort assumed in basic queuing theory), $C_X=1$. For a server with that service time distribution the RMWT is the average service time!

E.3 Application to Delayed Synchronous Immediate Requests

A delayed synchronous request is in a position somewhat like the position of someone who finds the taxi in the example is out. In this case, even though there may be multiple paths, CF engines and CFs, once the request is queued for a subchannel it waits for the current request on that subchannel to complete. In other words we can apply the theory above with reasonable confidence.

Example

We use data based on Figure 29 on page 124. For our purposes we do not need to worry about rates. We are only concerned with what happens if a synchronous request is delayed.

	REQ	AVG	STD_DEV
SYNC	687	425.6	208.7
ASYNC	224	2112.8	1749.1

The first step is to find the mean and standard deviation for the requests "as a whole." It is easy to see the overall average is 840 ms and it can be shown that the overall standard deviation is 1146.

Using the relations given in E.2, "Relation between Mean Square, Variance and Coefficient of Variation" we find that $C_x=1.86$ and hence that RMWT=1201 microseconds, rather than half the average service time (840/2=420 microseconds). In other words, synchronous requests that get delayed take about 3.8 times as long to complete as the average synchronous service time.

E.4 What about Queueing

You will notice that the forgoing really says nothing about queueing. How much queueing occurs will depend on the inter-arrival pattern of requests.

Appendix F. Special Notices

This redbook is intended to help IBM large systems customers and IBM systems professionals do capacity planning in a Parallel Sysplex environment. The information in this publication is not intended as the specification of any programming interfaces that are provided by CP2000. See the CP2000 tool under IBMLink/HONE/EHONE for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM′s product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM′s intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM (″vendor″) products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer′s ability to evaluate and integrate them into the customer′s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these

© Copyright IBM Corp. 1998

names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX	AIX/6000
APPN	AS/400
BatchPipes	BookManager
CICS	CICS/ESA
CICS/400	CICS/6000
CICSPlex	CUA
DB2	DFSMS
DFSMS/MVS	DFSORT
DRDA	Enterprise Systems Architecture/390
ES/3090	ES/9000
ESA/390	ESCON XDF
ESCON	Extended Services
GDDM	Hardware Configuration Definition
Hiperbatch	Hipersorting
IBM	IBMLink
IMS	IMS/ESA
MVS	MVS/ESA
MVS/SP	MVS/XA
NetReview	NetView
OPC	OS/2
OS/390	OS/400
Parallel Sysplex	PR/SM
Presentation Manager	PROFS
PS/2	QMF
RACF	Resource Measurement Facility
RISC System/6000	RMF
RS/6000	S/370
S/390	SNAP/SHOT
SP2	Sysplex Timer
System/370	System/390
Systems Validation Services	VM/ESA
VM/XA	VTAM
3090	400

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

Appendix G. Related Publications

At the time of writing, a few of these books are not available, but will be published in the near future.

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

G.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see “How to Get ITSO Redbooks” on page 309.

- *Accessing OpenEdition from the Internet*, SG24-4721
- *CICS and VSAM Record Level Sharing: Implementation Guide*, SG24-4766
- *DB2 for MVS/ESA V4 Data Sharing: Performance Topics*, SG24-4611
- *DB2 for MVS Locking*, SG24-4725
- *DB2 for OS/390 V5 Performance Topics*, SG24-2213
- *Introduction to Storage Performance Management Tools and Techniques for MVS/ESA*, GG24-4045 (available in softcopy only)
- *OpenEdition MVS for MVS/ESA 5.1 Presentation Guide*, GG24-4095 (available in softcopy only)
- *OS/390 MVS Parallel Sysplex Configuration, Volume 1: Overview*, SG24-2075 (not yet available)
- *OS/390 MVS Parallel Sysplex Configuration, Volume 2: Cookbook*, SG24-2076 (not yet available)
- *OS/390 MVS Parallel Sysplex Configuration, Volume 3: Connectivity*, SG24-2077 (not yet available)
- *OS/390 R3 OpenEdition, Installation and Customization*, SG24-2087 (not yet available)
- *Porting Applications to the OpenEdition MVS Platform*, GG24-4473
- *Selecting a Server - The Value of S/390*, SG24-4812
- *S/390 MVS Parallel Sysplex Performance*, SG24-4356

G.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
Application Development Redbooks Collection	SBOF-7290	SK2T-8037
Personal Systems Redbooks Collection	SBOF-7250	SK2T-8042

© Copyright IBM Corp. 1998

G.3 Other Publications

These publications are also relevant as further information sources:

G.3.1 CPSTOOLS Packages and Publications

Notice

These tools and documents are available to IBM employees and approved IBM Business Partners. The tools may be used with customers.

- *CP2000 Data Extraction Program*, CP2KEXTR package
- *CP2000 PC Tool*, CP2000 package
- *CP2000 S/390 Parallel Sysplex Quick-Sizer User's Guide*, SPSSZR package
- *CP2000 S/390 Parallel Sysplex Quick-Sizer*, SPSSZR package
- *Large Systems Performance Reference (LSPR)*, LSPRPC package
- *LPAR Capacity Estimator (LPAR/CE)*, LPARCE package
- *LPAR/CE User's Guide*, LPARCE package
- *LSPR/PC User's guide*, LSPRPC package
- *PCR Processor Capacity Reference, User's Guide*, PCR package
- *Processor Capacity Reference (PCR)*, PCR package

G.3.2 MKTTOOLS Packages and Publications

Notice

These tools and documents are available to IBM employees and may be used with customers.

- *Batch Workload Analysis Tool*, BWATOOL package
- *DB2 Family Client/Server Performance Series*, DB2PERF package
- *DB2 Extract Tool for Parallel Sysplex Input into CP2000*, DB2PARA package
- *IMS Guide for Locking Calculations*, PTSLOCK package
- *Large Systems Performance Reference (LSPR)*, LSPRPC package
- *OS/390 Performance Studies*, OS390PRF package
- *OS/390 Parallel Sysplex Test Report*, OS390T97 package
- *S/390 Parallel Sysplex Business Value*, BV390 package
- *S/390 Parallel Sysplex System Performance Considerations*, MFTS package
- *S/390 PSLC Software Pricer*, SWPRICER package
- *Parallel Sysplex Overhead: A Reality Check*, GF225009 package:
- *Software Migration Capacity Planning Aid*, SOFTCAP package

G.3.3 IBM Standard Publications

Notice

Books with an LY prefix are available to IBM-licensed customers only

- *Balanced Systems and Capacity Planning*, GG22-9299
- *CICS TS for OS/390 CICS Performance Guide*, SC33-1699
- *CICS TS for OS/390 System Definition Guide*, SC33-1682
- *DB2 for OS/390 V5 Administration Guide*, SC26-8957
- *DB2 for OS/390 V5 Data Sharing: Planning and Administration*, SC26-8961
- *DFSMS/MVS V1.3 DFSMSdfp Storage Administration Reference*, SC26-4920
- *DFSMS/MVS V1.3 Presentation Guide*, GG24-4391

- *ES/9000 and ES/3090 PR/SM Planning Guide*, GA22-7123
- *IBM SmartBatch for OS/390*, GC28-1627
- *IMS/ESA V5 Administration Guide: System*, SC26-8013
- *IMS/ESA V6 Administration Guide: Database Manager*, SC26-8725
- *IMS/ESA V6 Administration Guide: System*, SC26-8730
- *Large Systems Performance Reference (LSPR)*, SC28-1187
- *OpenEdition MVS Installation and Customization*, SG24-4529
- *OS/390 OpenEdition Planning*, SC28-1890
- *OS/390 RMF User's Guide*, SC28-1949
- *OS/390 RMF Report Analysis*, SC28-1950
- *OS/390 RMF Performance Management Guide*, SC28-1951
- *OS/390 RMF Programmer's Guide*, SC28-1952
- *OS/390 MVS Initialization and Tuning Guide*, SC28-1751
- *OS/390 MVS Initialization and Tuning Reference*, SC28-1752
- *OS/390 MVS Planning: Global Resource Serialization*, GC28-1759
- *OS/390 MVS Programming: Assembler Services Guide*, GC28-1762
- *OS/390 MVS Security Server System Programmer's Guide*, SC28-1913
- *OS/390 MVS Setting Up a Sysplex*, GC28-1779
- *OS/390 MVS System Commands*, GC28-1781
- *OS/390 Sysplex Application Migration*, GC28-1863
- *Performance Reporter for MVS V1R2.1 Capacity Planner Guide and Reference*, SH19-4021
- *S/390 MVS Sysplex Hardware and Software Migration*, GC28-1210
- *S/390 MVS Sysplex Overview: Introducing Data Sharing and Parallelism in a Sysplex*, GC28-1208
- *S/390 MVS Sysplex Systems Management*, GC28-1209
- *S/390 9672/9674/2003 PR/SM Planning Guide*, GA22-7236
- *S/390 9672/9674 Hardware Management Console Guide*, GC38-0453
- *S/390 9672/9674 Operations Guide, R2 and R3 Based Models*, GC38-3104
- *VTAM V4R3 Network Implementation Guide*, SC31-6548

G.3.4 Washington Systems Center Flashes

Notice

Washington Systems Center Flashes are available on IBMLink/HONE/EHONE. to IBM employees. Certain flashes may be available to registered customers on IBMLink/HONE/EHONE

- WSC Flash 9441: *MVS/ESA V5.1 Performance Information V5 Release To Release Migration Software Performance Impact*, OZSG023432
- WSC Flash 9505: *MVS Performance Capacity Planning Considerations For 9672-Rxx Processors*, OZSG023487
- WSC Flash 9510: *Expanded Storage is Still Very Important for DFSORT*, OZSG023510
- WSC Flash 9528: *Integrated Coupling Migration Facility (ICMF) Dispatching Enhancement*, OZSG023558
- WSC Flash 9609: *CF Reporting Enhancements To RMF V5.1*, OZSG023606
- WSC Flash 9609: *MVS/ESA Parallel Sysplex Performance LPAR Performance Considerations For Parallel Sysplex Environments*, OZSG023608
- WSC Flash 9613: *OS/390 Performance Information OS/390 Increased Virtual Storage Requirements*, OZSG023612

- WSC Flash 9723: *MVS/ESA Parallel Sysplex Performance XCF Performance Considerations*, OZSG023696
- WSC Flash 9731: *MVS/ESA Parallel Sysplex Performance Dynamic CF Dispatching*, OZSG023719
- WSC Flash 9715: *OS/390 R2 GRS Star APAR/PTF and Lock Structure Sizing*, OZSG023686

G.3.5 Other Books and Articles

- *Probability, Statistics, and Queueing Theory with Computer Science Applications*, A.O. Allen, Academic Press
- *Processor, I/O Path and DASD Configuration Capacity*, IBM Systems Journal, Vol. 20, No. 1,
- *The Cost of Scalability*, ITG International Technology Group
 http://www.s390.ibm.com/marketing/costs.html

G.4 How to Reach CP2000, CPAIP, LPAR/CE, LSPR/PC, PCR and Quick-Sizer

For customer access to CP2000 or Quick-Sizer, please have your IBM representative contact the CP2000 hotline *DALVM41B(CP2000)*.

CP2000 forums are available on IBMMVS.

Packages are available from CPSTOOLS. If CPSTOOLS is available on your system, the packages are obtained using any of the following commands:

- TOOLCAT CPSTOOLS
- OMNIDISK CPSTOOLS GET *name* PACKAGE
- OMNIDISK CPSTOOLS SUBSCRIBE *name* PACKAGE

If CPSTOOLS is not available on your system, the packages are obtained using the following command:
TOOLS SENDTO WSCVM TOOLS CPSTOOLS GET *name* PACKAGE

For *name*, substitute the package name, for example:

OMNIDISK CPSTOOLS GET PCR PACKAGE
or
TOOLS SENDTO WSCVM TOOLS CPSTOOLS GET PCR PACKAGE
Processor Capacity Reference (PCR).

LPARCE:
LPAR Capacity Estimator (LPAR/CE).

LSPRPC:
Large Systems Performance Reference (LSPR).

SPSSZR:
CP2000 S/390 Parallel Sysplex Quick-Sizer.

Note: Besides GET, SUBSCRIBE and INFORM, requests are also honored to receive a catalog of CPSTOOLS packages that are available. Issue a GET for CPSTOOLS CATALOG.

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at http://www.redbooks.ibm.com.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type GOPHER.WTSCPOK.ITSO.IBM.COM
- **Tools disks**

 To get LIST3820s of redbooks, type one of the following commands:

  ```
  TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
  TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
  ```

 To get BookManager BOOKs of redbooks, type the following command:

  ```
  TOOLCAT REDBOOKS
  ```

 To get lists of redbooks, type one of the following commands:

  ```
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
  ```

 To register for information on workshops, residencies, and redbooks, type the following command:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1998
  ```

 For a list of product area specialists in the ITSO: type the following command:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
  ```

- **Redbooks Web Site on the World Wide Web**

 http://w3.itso.ibm.com/redbooks

- **IBM Direct Publications Catalog on the World Wide Web**

 http://www.elink.ibmlink.ibm.com/pbl/pbl

 IBM employees may obtain LIST3820s of redbooks from this page.

- **REDBOOKS category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

 With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (http://www.redbooks.ibm.com/redpieces.htm). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

© Copyright IBM Corp. 1998

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** — send orders to:

	IBMMAIL	**Internet**
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications	IBM Publications	IBM Direct Services
Publications Customer Support	144-4th Avenue, S.W.	Sortemosevej 21
P.O. Box 29570	Calgary, Alberta T2P 3N5	DK-3450 Allerød
Raleigh, NC 27626-0570	Canada	Denmark
USA		

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States)** or **(+1)001-408-256-5422 (Outside USA)** — ask for:

 Index # 4421 Abstracts of new redbooks
 Index # 4422 IBM redbooks
 Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com
- **On the World Wide Web**

Redbooks Web Site	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Internet Listserver**

 With an Internet e-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an e-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank).

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (http://www.redbooks.ibm.com/redpieces.htm). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

- Invoice to customer number
- Credit card number

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Glossary

Explanations of Cross-References: The following cross-references are used in this glossary:

Contrast with. This refers to a term that has an opposed or substantively different meaning.

See. This refers the reader to multiple-word terms in which this term appears

See also. This refers the reader to terms that have a related, but not synonymous meaning.

A

active service policy. The service policy that determines workload management processing if the Sysplex is running in goal mode. See *goal mode*.

affinity. (1) A connection or association between two objects. (2) A relationship between a LU, a generic resource, and a specific instance of a resource

alternative wait management (AWM). Alternate Wait Management support, available since MVS V4.3, enhances the management of a logical CP entry into wait when running in LPAR mode. The synergistic relationship between the dispatchers in OS/390 and LPAR significantly reduces the frequency of LPAR dispatches. The result is a major reduction in LPAR low utilization effects. The greatest reductions are for partitions with more than one logical CP. See also *low utilization effect*.

analog. Pertaining to data consisting of continuously variable physical quantities.

application. A collection of software components used to perform specific types of work on a computer; for example, a payroll application or an airline reservation application.

architecture. A logical structure that encompasses operating principles including services, functions, and protocols. See *computer architecture*.

ARIAS. See *CPAIP*.

asynchronous. Without regular time relationship. Unexpected or unpredictable with respect to the program′s instructions, or to time. Contrast with *synchronous*.

B

base sysplex. The set of one or more systems that is given a cross-system coupling facility (XCF) name, and in which the authorized programs can then use XCF coupling services. A base sysplex does not include a CF. See also *Parallel Sysplex* and *sysplex*.

basic configuration unit (BCU). A set of DASD connected to the same control unit. Each BCU is specified by a control unit type, DASD type, and number of connected DASD.

basic mode. A central processor mode that does not use logical partitioning. Contrast with *logically partitioned (LPAR) mode*.

batch. A program or operation that is performed with little or no interaction between the user and the system.

buffer. (1) A portion of storage used to hold input or output data temporarily. (2) A routine or storage used to compensate for a difference in either data rate or time of occurrence of events, when transferring data from one device to another.

buffer invalidation. A technique for preventing the use of invalid data in a Parallel Sysplex data sharing environment. The technique involves marking all copies of data in DB2 or IMS buffers invalid once a sharing DBMS subsystem has updated that data.

buffer pool. A set of buffers that contains buffers of the same length.

business element. Several business elements make up the dimensioning workload. Business elements are grouped based on different business applications such as:

- Back office application
- Teller application
- ATM application

Business elements can also be grouped based on their technical infrastructure such as:

- CICS
- TSO

See also *workload* and *dimensioning workload*.

business management. Business management provides inventory management, security management, service level agreement planning and control, financial administration, business planning, and management services for all enterprise-wide computer-related facilities. See also *system management*.

© Copyright IBM Corp. 1998

C

cache. A random access electronic storage in selected storage controls used to retain frequently used data for faster access by the channel.

cache structure. A CF structure that enables high-performance sharing of cached data by multisystem applications in a sysplex. Applications can use a cache structure to implement several different types of caching systems, including a store-through or a store-in cache.

cache structure services. OS/390 services that enable applications in a sysplex to perform operations such as the following on a CF cache structure:

- Manage cache structure resources
- Store data into and retrieve data from a cache structure
- Manage accesses to shared data
- Determine when shared data has been changed
- Determine whether a local copy of shared data is valid.

capacity. (1) The ability to hold, receive, store, or accommodate. (2) A measure of content: ability to contain. (3) Maximum production or output.

capture ratio. The ratio of measured to adjusted CPU utilization. The workload CPU is adjusted to distribute uncaptured CPU and other CPU (system, VTAM, monitors, and so forth) over the business units. The value is usually between 0.5 and 1.0. In RMF, the percent of CPU time accounted for is in the Workload Activity Report. It is derived from other calculated values. The capture ratio is calculated as:

$$\text{capture ratio} = \frac{100 \times (\text{ALL TCB seconds} + \text{SRB seconds per CP})}{\text{CPU seconds per CP}}.$$

CBW2. The LSPR (Commercial Batch Workload 2) workload is a commercial batch jobstream. It consists of 32 jobs, with 157 job steps. These jobs are more resource-intensive than jobs in the CB84 workload, use more current software, and exploit ESA features. The work done by these jobs includes various combinations of C, COBOL, FORTRAN, and PL/I compile, link-edit, and execute steps. Sorting, DFSMS utilities (for example dump/restore and IEBCOPY), VSAM and DB2 utilities, SQL processing, SLR processing, GDDM graphics, and FORTRAN engineering/scientific subroutine library processing are also included.

Compared to CB84, there is much greater use of JES processing, with more JCL statements processed and more lines of output spooled to the SYSOUT and HOLD queues. This workload is heavily DB2-oriented, with about half the processing time performing DB2-related functions. See also *LSPR*, *CB84* and *FPC1*.

CB84. The LSPR CB84 workload is a moderate commercial batch jobstream consisting of 130 jobs, with 610 unique job steps. The work done by these jobs include various combinations of compile, link-edit, and execute steps. Utility jobs, primarily for data manipulation, are also included. Measurements are made with ESA levels of MVS/SP, DFSMS, JES2, and RMF. Assembler H, COBOL/VS, PL/I Optimizing Compiler, and DFSORT software is also used by the jobstream. Access methods include BSAM, QSAM, BDAM, an VSAM. SMS is used to manage non-system data. Performance data collected consists of the usual SMF data, including type 30 records (workload data and RMF data). See also *LSPR*, *CBW2* and *FPC1*.

central electronic complex (CEC). The combination of processors (CPs), processor storage, and channel subsystem which is installed as a unit. In CP2000, if the CEC is single-system image, the CEC is identical to the system. If the CEC is used to run logical partitions, each system image is treated separately. A physically partitioned machine should be treated as two CECs. See also *central processor complex, central processing unit, central processor, logical partition, physical partition,* and *physically partitioned configuration.*

central processing unit (CPU). The part of a computer that includes the circuits that control the interpretation and execution of instructions.

central processor (CP). The part of the computer that contains the sequencing and processing facilities for instruction execution, initial program load, and other machine operations.

central processor complex (CPC). A physical collection of hardware that includes central storage, one or more central processors, Sysplex Timers, and channels. See also *central electronic complex*.

central storage. Storage that is an integral part of the processor unit. Central storage includes both main storage and the hardware system area (HSA).

change management. Change management helps the user coordinate enhancements and fixes the software and hardware components, data-processing procedures, application programs, and other facilities of the data-processing installation. It provides management with the information that will enable them to make better decisions about the frequency, timing, and types of changes planned for the systems environment in order to minimize disruption of service. See also *system management*.

channel. (1) A functional unit, controlled by a host computer, that handles the transfer of data between processor storage and local peripheral equipment. (2) A path along which signals can be sent. (3) The portion of a storage medium that is accessible to a given reading or writing station. (4) In broadband

transmission, a designation of a frequency band 6 MHz wide.

channel operation. The term channel operation is related to DASD subsystem statistics. A channel operation is defined to start at a LOCATE RECORD or SEEK command and continue until the next LOCATE RECORD or SEEK command, or to the end of the CCW chain.

channel program. A channel program consists of one or more CCWs that are arranged for sequential execution. The channel program is started by the START SUBCHANNEL instruction.

CICSPlex. (1) The largest set of CICS systems to be monitored and controlled as a single entity. (2) In a CICSPlex SM environment, the user-defined name, description, and configuration information for a CICSPlex. A CICSPlex can be made up of CICS systems or CICS system groups.

cloning. Creating multiple, (almost) identical systems.

coefficient of variation, squared.

$$C^2 = \frac{STD^2}{AVG^2}$$

Where STD = standard deviation and AVG = average (mean). Coefficient of variation tells directly how far away the distribution is from the exponential distribution. C^2 for the exponential distribution is one. If C^2 is larger than one, the distribution is hyperexponential (clustered). If C^2 is less than one, the distribution is more condensed (less spread, less clustered). C^2 is zero for constant values.

computer architecture. The organizational structure of a computer system, including hardware and software.

configuration. The arrangement of a computer system or network as defined by the nature, number, and chief characteristics of its functional units. More specifically, the term *configuration* may refer to a hardware configuration or a software configuration. See also *system configuration*.

control interval (CI). A fixed-length area of direct access storage in which VSAM stores records and creates distributed free space. Also, in key-sequenced data set or file, the set of records pointed to by an entry in the sequence-set index record. The control interval is the unit of information that VSAM transmits to or from direct access storage. A control interval always comprises an integral number of physical records.

Coupling Facility (CF). A logical partition (LPAR) that provides high-speed caching, list processing, and locking functions in a Parallel Sysplex.

Coupling Facility channel (coupling link). A high bandwidth fiber-optic channel that provides the high-speed connectivity required for data sharing between a CF and the central processor complexes directly attached to it.

Coupling Facility white space. Coupling Facility storage set aside for rebuilding of structures from other CFs, in case of failure.

coupling services. In a sysplex, the functions of XCF that transfer data and status between members of a group residing on one or more systems in the sysplex.

CPU service unit. A measure of the task control block (TCB) execution time multiplied by an SRM constant which is CPU model-dependent. Included in the execution time is the time used by the address space while executing in cross-memory mode (that is, during either secondary addressing mode or a cross-memory call). This execution time is not counted for the address space that the target of the cross-memory reference. See also *service unit*, *SRB service unit*, *I/O service unit*, *storage service unit* and *service definition coefficients*.

CP2KEXTR (CP2000 extract). A program which can be run on an OS/390 system to build a file for auto input in CP2000. You would run this program, then download the file to a diskette, and finally upload the file to a PC if using CP2000. Note that CP2000 is available on IBMLink for approved IBM Business Partners.

Customer Information Control System (CICS). An IBM-licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs. It includes facilities for building, using, and maintaining databases.

D

database. (1) A set of data, or part or the whole of another set of data that consists of at least one file, and that is sufficient for a given purpose or for a given data-processing system. (2) A collection of data fundamental to a system.

Data Language/I (DL/I). The IMS data manipulation language, a common high-level interface between a user application and IMS. DL/I calls are invoked from application programs written in languages such as PL/I, COBOL, VS Pascal, C, and Ada. It can also be invoked from assembler language application programs by subroutine calls. IMS lets the user define data structures, relate structures to the application, load structures, and reorganize structures.

data sharing. In a Parallel Sysplex, the ability of concurrent subsystems (such as DB2 or IMS database managers) or application programs to directly access and change the same data while maintaining data integrity. See also *Sysplex data sharing*.

data sharing group. See also *DB2 data sharing group* and *IMS/DB data sharing group*.

DB2 data sharing group. A collection of one or more concurrent DB2 subsystems that directly access and change the same data while maintaining data integrity.

destage. The asynchronous write of new or updated data from cache or nonvolatile storage to DASD.

device. (1) A mechanical, electrical, or electronic contrivance with a specific purpose. (2) An input/output unit such as a terminal, display, or printer.

dimensioning workload. In capacity planning, the part of the workload that is planned for. The dimensioning workload is usually comprised of a number of business elements. See also *workload* and *business element*.

direct access storage device (DASD). A device in which access time is effectively independent of the location of the data.

E

EMIF. ESCON multiple image facility.

enclave. Group of associated dispatchable units. More specifically, a group of SRB routines that are to be managed and reported on an entity. They can be created, for example, by distributed DB2. Enclaves are represented by different SRBs in different address spaces.

Enterprise Systems Connection (ESCON). A set of products and services that provides a dynamically connected environment using optical cables as a transmission medium.

ESA/390. Enterprise Systems Architecture/390.

Enterprise Systems Connection (ESCON). A set of products and services that provides a dynamically connected environment using optical cables as a transmission medium.

ESCON channel. A channel having an Enterprise Systems Connection channel-to-control-unit I/O interface that uses optical cables as a transmission medium.

ESCON Director (ESCD). A device that provides connectivity capability and control for attaching any two links to each other.

ESCON Extended Distance Feature (ESCON XDF). An ESCON feature that uses laser/single mode fiber optic technology to extend unrepeated link distances up to 20 km.

ESCON multiple image facility (EMIF). A facility that allows channels to be shared among PR/SM logical partitions in an ESCON environment.

execution velocity. A service goal naming the rate at which you expect work to be processed for a given service class, or a measure of the acceptable processor and storage delays while work is running. Execution velocity factors in the I/O delays, buffer pool delays, and server environment delays.

expanded storage. (1) Optional integrated high-speed storage that transfers 4K-byte pages to and from central storage. (2) Additional (optional) storage that is addressable by the system control program.

exponential distribution. When a distribution is random and memoryless, it is said to be exponential. This is described by the formula:

$$F(x) = 1 - e^{-\frac{x}{a}}; \text{ where } x > 0$$

Where a is the average of the random variable x. Exponential distribution is very useful as it can be expressed and calculated in simple ways in mathematics. It also very often fits to the distributions existing in the real world, like the interarrival time distribution of the arriving transactions to the CEC.

F

forecast. (1) To plan ahead. (2) Calculate or predict (some future event or condition) usually as a result of rational study and analysis of available pertinent data. (3) To indicate as likely to occur.

FPC1. The FPC1 (Floating Point Commercial 1) LSPR batch workload is an engineering and manufacturing batch jobstream, representative of scientific data development work. It includes jobs doing static analysis, dynamic analysis, computational fluid dynamics, nuclear fuel calculations, and circuit analysis. Being scientific work, it makes heavy use of floating point. While FPC1 is a processor-intensive workload, it also places significant demands on external resources such as central storage, channels, and DASD. See also *LSPR*, *CBW2*, and *CB84*.

G

generic resource (GR). A name representing multiple identical instances of a resource.

G/G/C. The specification of a queueing system. The first G says that the interarrival time distribution of the transactions is general (not any specific like exponential distribution). The second G says that the service time distribution for the transactions is general. The letter C gives the number of servers. See also *M/M/C*.

gigabytes. One billion (10•) bytes.

goal mode. A mode of processing where the active service policy determines system resource management.

GRS. Global Resource Serialization, part of the OS/390 base control program. Global resource serialization offers the control needed to ensure the integrity of resources in a multisystem environment. Combining the systems that access shared resources into a global resource serialization complex serializes access to resources across multiple systems. A global resource serialization complex consists of one or more systems connected by communication links.

GRS ring. In a GRS ring complex, all global resource requests are processed by all systems in the ring. As part of the processing, each system checks for global resource contention when adding a new request to its complex-wide global resource queue.

GTF trace. The Generalized Trace Facility (GTF) is a function provided in the OS/390 operating system to trace system events for diagnostic purposes. System events include dispatching tasks, Start Subchannel requests supervisor calls (SVCs), I/O interrupts, and so on.

H

hardware system area (HSA). A logical area of central storage, not addressable by application programs, used to store Licensed Internal Code and control information.

highly parallel. Refers to multiple systems operating in parallel, each of which can have multiple processors. See also *n-way*.

high-speed buffer. A cache or a set of logically partitioned blocks that provides significantly faster access to instructions and data than provided by central storage.

host (computer). (1) In a computer network, a computer that provides end users with services such as computation and databases and that usually performs network control functions. (2) The primary or controlling computer in a multiple-computer installation.

HSA. Hardware system area.

I

IBMLink. IBMLink is a single framework offering interactive applications, electronic mail, electronic data interchange and file transfer (IBM)

IMS/DB data sharing group. A collection of one or more concurrent IMS/DB subsystems that directly access and change the same data while maintaining data integrity. The components in an IMS/DB data sharing group include the sharing IMS subsystems, the IRLMs they use, the IRLM, OSAM, and VSAM structures in the CF, and a single set of DBRC RECONs.

IMS TM. Information Management System Transaction Manager.

Information Management System (IMS). A general purpose system whose full name is Information Management System/Virtual Storage (IMS/VS).

input/output (I/O). (1) Pertaining to a device whose parts can perform an input process and an output process at the same time. (2) Pertaining to a functional unit or channel involved in an input process, output process, or both, concurrently or not, and to the data involved in such a process.

input/output channel. In a data processing system, a functional unit, controlled by the processing unit, that handles the transfer of data between main storage and peripheral equipment.

intensity. From Little′s Law, intensity is the product of arrival rate and response time. It is the average number in the system or total time for those users.

internal throughput rate (ITR). A measurement of the number of transactions processed per processor busy second. The ITR is calculated by dividing the number transactions per second by the product of the average CPU busy and the number of CPs. See also *ITRR* and *LSPR*.

internal throughput rate ratio (ITRR). It is the ratio of the ITR of two CECs.

invalidation. The process of removing records from cache because of a change in status of a subsystem facility or function, or because of an error while processing the cache image of the set of records. When such a cache image is invalidated, the corresponding records cannot be accessed in cache and the assigned cache space is available for allocation.

I/O intensity. The average I/O time is the expected value of one I/O. Intensity is the expected time per second. So if an I/O takes 10 ms. on average (queuing and service) and the rate per second is 20, the intensity is 20 multiplied by 10 or 200 ms. This is an indicator of total I/O time.

I/O service unit. Measurement of individual data set I/O activity and JES spool reads and writes for all data sets associated with the address space. SRM calculates I/O service using either I/O block (EXCP) counts or device connect time (DCTI), as specified on the IOSRVC keyword in the IEAIPSxx parmlib member. If DCTI is used to calculate I/O service, operations to VIO data sets and to devices that the channel measurement facility does not time are not included in the I/O service total.

When an address space executes in cross-memory mode (that is, during either secondary addressing mode or a cross-memory call), the EXCP counts or the DCTI will be included in the I/O service total. This I/O service is not counted for the address space that is the target of the cross memory reference. See also *service unit*, *SRB service unit*, *CPU service unit*, *storage service unit* and *service definition coefficients*.

ITR. See *internal throughput rate*.

ITRR. See *internal throughput rate ratio*.

J

JES (Job Entry Subsystem). A system facility for spooling, job queuing, and managing I/O.

L

latency. The time interval between the instant at which an instruction control unit initiates a call for data and the instant at which the actual transfer of data starts.

latent demand. Work that is not being submitted to the system or is considerably delayed due to some constraint. Work is not being done or is completed in an untimely fashion. Assessing the size of the latent demand is impossible.

Licensed Internal Code (LIC). Software provided for use on specific IBM machines and licensed to customers under the terms of IBM′s Customer Agreement. Microcode can be Licensed Internal Code and licensed as such.

link. The combination of physical media, protocols, and programming that connects devices.

list structure. A CF structure that enables multisystem applications in a Sysplex to share information organized as a set of lists or queues. A list structure consists of a set of lists and an optional lock table, which can be used for serializing resources in the list structure. Each list consists of a queue of list entries.

list structure services. OS/390 services that enable multisystem applications in a sysplex to perform operations such as the following on a CF list structure:

- Read, update, create, delete, and move list entries in a list structure.
- Perform serialized updates on multiple list entries in a list structure.
- Monitor lists in a list structure for transitions from empty to non-empty.

local area network (LAN). A data network located on the user′s premises in which serial transmission is used for direct data communication among data stations. It services a facility without the use of common carrier facilities.

local cache. A buffer in local system storage that might contain copies of data entries in a CF cache structure.

lock contention. The situation when a locking request cannot be satisfied because the lock entry is already occupied for another locking request. Two types of lock contention can be distinguished:

1. True lock contention (two or more requesters show interest in the same data item)
2. False lock contention (two or more requesters show interest in different data items, but the hashing algorithm maps these requests to the same lock entry)

lock resource. Data accessed through a CF structure.

lock structure. A CF structure that enables applications in a sysplex to implement customized locking protocols for serialization of application-defined resources. The lock structure supports shared, exclusive, and application-defined lock states, as well as generalized contention management and recovery protocols.

lock structure services. OS/390 services that enable applications in a sysplex to perform operations such as the following on a CF lock structure:

- Request ownership of a lock.
- Change the type of ownership for a lock.
- Release ownership of a lock.
- Manage contention for a lock.
- Recover a lock held by a failed application.

logically partitioned (LPAR) mode. A central processor complex (CPC) power-on reset mode that enables use of the PR/SM feature and allows an operator to allocate CPC hardware resources (including central processors, central storage,

expanded storage, and channel paths) among logical partitions. Contrast with *basic mode*.

logical partition (LP). In LPAR mode, a subset of the processor unit resources that is defined to support the operation of a system control program (SCP). See also *logically partitioned (LPAR) mode*.

loosely coupled. A multisystem structure that requires a low degree of interaction and cooperation between multiple OS/390 images to process a workload. See also *tightly coupled*.

low utilization effect (LUE). Low utilization effect may be described as a unit of work requiring more processor busy time at low utilizations than at high utilizations. The effect is that reported CPU utilizations do not necessarily reflect available system capacity. Reported CPU utilizations are accurate for the reporting intervals; however, care should be exercised when drawing capacity conclusions from the reported values. See also *alternative wait management*.

LP. Logical partition.

LPAR. Logically partitioned (mode).

LPAR/CE. LPAR/CE is a PC-based (DOS) productivity tool designed to assess IBM processor capacity when running various workload types in independent logical partitions under LPAR. The tool is based on IBM′s LSPR data and LPAR benchmark tests. See also *LPAR* and *LSPR*.

LSPR. Large Systems Performance Reference is a set of IBM-conducted measurements. These measurements are done in an unconstrained environment, at a well-defined level of CP load, and using well-defined workloads at well-defined operating system levels. See also *LSPR ratio*, *LSPR/PC*, *PCR*. *M-value* and *RPP*.

LSPR/PC. LSPR/PC is a PC-based (DOS) productivity tool, designed to provide relative capacity assessments for System 370/390 architecture processors. The tool is based on IBM′s LSPR Internal Throughput Rate (ITR) data. See also *LSPR*, *LSPR ratio*, *ITR*, *M-value*, *RPP* and *PCR*.

LSPR ratio. The IBM LSPR ratios represent IBM′s assessment of relative processor capacity in an unconstrained environment for the specific benchmark workloads and system control programs specified in the LSPR tables. Ratios are based on measurements and analysis. The amount of analysis as compared to measurement varies with each processor. Some non-IBM processor ratios are based on vendor performance claims. When vendor performance claims are used to establish the ratio, the source of the vendor claim is identified in the tables. See also *LSPR*, *LSPR/PC*, *PCR*, *M-value* and *RPP*.

LUE. See *low utilization effect*

M

M-value. A measure of relative processor capacity based on ITR ratios. For OS/390, the ITR is based upon the harmonic average of equal parts of TSO, IMS/DS, and CICS. The base model used to generate an ITRR is the 3090-180E. It varies by SCP, processor model, LPAR configuration, and workload. See also *internal throughput rate* and *internal throughput rate ratio*. M-values are calculated using the Harmonic Mean formula:

$$M\text{-}value = \frac{1}{\frac{1/3}{TSO.ITRR} + \frac{1/3}{CICS.ITRR} + \frac{1/3}{IMS.ITRR}}$$

The M value for the machine scales to the ITRR as follows. $M\text{-}value = ITRR \times 744.35$.

An approximation for a MIPS value for a machine can be obtained as follows. $MIPS = \frac{M\text{-}value}{43}$

mainframe. A large computer, in particular one to which other computers can be connected so that they can share facilities the mainframe provides; for example, a System/390 computing system to which personal computers are attached so that they can upload and download programs and data.

main storage. A logical entity that represents the program addressable portion of central storage. All user programs are executed in main storage. See also *central storage*.

massively parallel. Refers to thousands of processors in a parallel arrangement.

MDF. See *multiple domain facility*.

memory. Program-addressable storage from which instructions and other data can be loaded directly into registers for subsequent execution or processing. Synonymous with *main storage*.

microcode. (1) One or more microinstructions. (2) A code, representing the instructions of an instruction set, that is implemented in a part of storage that is not program-addressable. (3) To design, write, and also test one or more microinstructions.

microprocessor. A processor implemented on one or a small number of chips.

migration. Installing a new version or release of a program when an earlier version or release is already in place.

MIPS. (millions of instructions per second). Literally, the speed on which a CP on a CEC executes instructions from its own instruction set. Therefore

MIPS cannot be used to describe the relative power of the CEC, because the instruction sets, the speeds of the instructions and the systems software are different in different CECs. Still, MIPS is used to describe the relative power of the CEC, but then the one who is using MIPS commonly forgets to mention the application set, which has a strong effect to the relative power of the CECs. See also RIPS.

mixed complex. A global resource serialization complex in which one or more of the systems in the global resource serialization complex are not part of a multisystem sysplex.

MLPF. See *multiple logical processor facility*.

M/M/C. The specification of a queueing system using Kendal notation. The first M comes from the name Markow, and it says that the interarrival time distribution of the transactions is exponential. The second M comes from the name Markow as well, and it says that the service time distribution for the transactions is exponential. The letter C gives the number of servers, for example the number of the CPs in a CEC. Additionally M/M/C assumes (defaults) that the source of the transactions (the population generating transactions) is infinite (in practice very large). And it assumes that the size of the queue is infinite (in practice very large). Additionally assumed is FCFS (First Come First Served) queue discipline. See also *G/G/C*.

MP. See *multiprocessor*.

multifiber cable. An optical cable that contains two or more fibers.

multimode optical fiber. A graded-index or step-index optical fiber that allows more than one bound mode to propagate. Contrast with *single mode optical fiber*.

multiple domain facility (MDF). The LPAR implementation from Amdahl Corp.

multiple logical processor facility (MLPF). The LPAR implementation from Hitachi Corp.

multi-OS/390 environment. An environment that supports more than one OS/390 image. See also *OS/390 image* and *sysplex*.

multiprocessing. The simultaneous execution of two or more computer programs or sequences of instructions. See also *parallel processing*.

multiprocessor (MP). A CPC that can be physically partitioned to form two operating processor complexes.

multisystem sysplex. A sysplex in which two or more OS/390 images are allowed to be initialized as part of the sysplex. See also *single-system sysplex*.

N

n-way. The number *(n)* of CPs in a CEC. For example, a 6-way CEC contains six CPs.

network. A configuration of data processing devices and software connected for information interchange.

O

online. Pertaining to equipment, devices, or data under the direct control of the processor.

operating system (OS). Software that controls the execution of programs and that may provide services such as resource allocation, scheduling, input/output control, and data management. Although operating systems are predominantly software, partial hardware implementations are possible. Examples are IBM PC DOS, IBM OS/2 and OS/390.

operations management. Operations management manages the use system resources to support enterprise information processing workloads. It is the setting of policies to manage workloads and resource availability, and the automation of the interactions required to implement these policies. See also *system management*.

optical cable. A fiber, multiple fibers, or a fiber bundle in a structure built to meet optical, mechanical, and environmental specifications.

OS/390 image. A single occurrence of the OS/390 operating system that has the ability to process work.

OS/390 system. An OS/390 image together with its associated hardware, which collectively are often referred to simply as a system, or OS/390 system.

P

PAR. See *peak-to-average ratio*.

parallel. (1) Pertaining to a process in which all events occur within the same interval of time, each handled by a separate but similar functional unit; for example, the parallel transmission of the bits of a computer word along the lines of an internal bus. (2) Pertaining to concurrent or simultaneous operation of two or more devices or to concurrent performance of two or more activities in a single device. (3) Pertaining to concurrent or simultaneous occurrence of two or more related activities in multiple devices or channels. (4) Pertaining to the simultaneity of two or more processes. (5) Pertaining to the simultaneous processing of the individual parts of a whole, such as the bits of a character and the characters of a word, using separate facilities for the various parts. (6) Contrast with *serial*.

parallel processing. The simultaneous processing of units of work by many servers. The units of work can be either transactions or subdivisions of large units of work (batch).

Parallel Sysplex. A Parallel Sysplex is a sysplex with one or more CFs. See also *base sysplex* and *sysplex*. CP2000 sometimes uses the term SPTS for a Parallel Sysplex. See *S/390 Parallel (transaction) Sysplex*.

partition. An area of storage on a fixed disk that contains a particular operating system or logical drives where data and programs can be stored.

partitionable CPC. A CPC can be divided into two independent CPCs. See also *physical partition*, *MP*, and *side*.

PCR. The CP2000 Processor Capacity Reference is a PC-based (OS/2) productivity tool intended to assess relative processor capacity within the various processor families of System 370/390. The tool is based on IBM′s LSPR data. See also *LSPR*, *LSPR/PC* and *LSPR ratio*.

peak. (1) The highest level or greatest degree. (2) A high point in a course of development, especially as represented on a graph.

peak-to-average ratio (PAR). A measure of the variation of CPU utilization obtained by dividing the peak sample from a set of measurement intervals (usually an hour) by the average of all the samples.

performance. For a storage subsystem, a measurement of effective data processing speed against the amount of resource that is consumed by a complex. Performance is largely determined by throughput, response time, and system availability.

performance block. A piece of storage containing workload management′s record of execution delay information about work requests.

performance management. (1) The process workload management uses to decide how to match resources to work according to performance goals and processing capacity. (2) Performance management addresses the effectiveness with which information systems deliver service to their users. See also *system management*.

performance period. A service goal and importance level assigned to a service class for a specific duration. You define performance periods for work that has variable resource requirements.

persistent connection. A connection to a CF structure with a connection disposition of KEEP. OS/390 maintains information about the connection so that when the connection terminates abnormally from a CF structure, OS/390 places the connection in a failed-persistent state, and the connection can attempt to reconnect to the structure.

persistent structure. A structure allocated in the CF with a structure disposition of KEEP. A persistent structure keeps its data intact across system or sysplex outages, regardless of whether any users are connected to the structure.

physical partition. Part of a CPC that operates as a CPC in its own right, with its own copy of the operating system.

physically partitioned (PP) configuration. A system configuration that allows the processor controller to use both CPC sides as individual CPCs; the A-side of the processor controls side 0, the B-side controls side 1. Contrast with *single-image (SI) mode*.

policy. A set of installation-defined rules for managing sysplex resources. The XCF PR/SM policy and sysplex failure management policy are examples of policies.

power-on reset. The state of the machine after a logical power-on before the control program is IPLed.

preference list. An installation list of CFs, in priority order, that indicates where OS/390 is to allocate a structure.

problem management. The process of managing problems and incidents from their detection through their final resolution. It encompasses the detection, analysis, recovery, resolution, and tracking of problems and incidents that occur. See also *system management*.

processing unit. The part of the system that does the processing, and contains processor storage.

processor. A processing unit, capable of executing instructions when combined with main storage and channels.

processor complex. A physical collection of hardware that includes main storage, one or more processors, and channels.

Processor Resource/Systems Manager (PR/SM). A function that allows the processor unit to operate several system control programs simultaneously in LPAR mode. It provides for logical partitioning of the real machine and support of multiple preferred guests. See also *LPAR.*

protocol. A specification of the format and relative timing of information exchanged between peer entities within a layer.

Q

queuing. Programs waiting, in order determined by their priority, for access to the central processor.

Quick-Sizer. See *SPSSZR*.

R

record. A set of data treated as a unit.

relative I/O content (RIOC). The ratio of the total number of I/O operations *(S)* per second to the total power used for the same period of time ($M \times B$). It provides a way of describing a workload in terms of I/O content, where a higher number indicates a workload of higher I/O content. It is calculated by the following formula: $RIOC = \frac{S}{(M \times B)}$ where M is the power of the CPU and B is the proportion used.

relative processor power (RPP). A unit used to express processor capacity. RPP is a measured average of well-defined workload profiles ITR-ratios. Internal Throughput Rate (ITR) is measured in transactions/CPU second. Large Systems Performance Reference (LSPR) measurements predict RPP values for processors running certain releases of operating systems. See also *ITR*, *M-value*, *LSPR*, *LSPR/PC*, and *PCR*.

report class. A group of work for which reporting information is collected separately. For example, you can have a WLM report class for information combining two different service classes, or a report class for information on a single transaction.

request. A service primitive issued by a service user to call a function supported by the service provider.

resource. Any facility of a computing system or operating system required by a job or task. This includes storage, processor, channels, volumes or processing programs.

resource group. An amount of processing capacity across one or more OS/390 images, assigned to one or more WLM service classes.

response time. (1) Transaction response time is the amount of time it takes after a user presses the enter key at the terminal until the reply appears at the terminal. (2) Device service time is device service time plus software queuing (IOS queue time) for the request. Software queuing comes about because of multiple requests for data on the same actuator - one request will be serviced while the rest wait. Response time is considered more meaningful than device service time since this is the actual time the application requesting service sees. Response time is measured in milliseconds.

RIOC. See *relative I/O content*.

RIPS. (relative instructions per second). RIPS is used to compare the processing power of different CECs. Here, you mention the particular CEC that is the reference point and its RIPS-value. Further, you have to describe the workload mix used and the operating system used. The starting RIPS value is selected arbitrarily (for example, close to the MIPS value claimed by you or a consultant) *if the RIPS values for other CECs are scaled correctly.* This should be based on LSPR measurements, as it is a reliable source for comparisons, widely accepted inside and outside IBM. One advantage of using RIPS is that the path lengths for online transactions are easily tackled because they fit into this environment as well. See also MIPS.

RMF. Resource Measurement Facility (RMF) is a measurement collection tool that is designed to measure selected areas of OS/390 system activity and present the data in the form of system management facility (SMF) records, formatted printed reports, formatted display reports, workstation based graphical reports, and reports formatted by spreadsheet applications.

RMF Postprocessor. A function of RMF which generates formatted printed reports from RMF SMF input data.

RMF Spreadsheet Report. A function of RMF which uses workstation-based spreadsheet applications to generate graphics and reports from data output of the RMF Postprocessor.

routing. The assignment of the path by which a message will reach its destination.

RPP. See *relative processor power*.

S

saturation design point (SDP). A user-defined utilization level, expressed as a percentage of the capacity of the system image or CEC, above which it is believed that user response times will be noticeably increased. Using the peak-to-average ratio (PAR), it is calculated as 100/PAR. This means that when the SDP is reached, the peaks will be at 100%. See also *peak-to-average ratio*.

serial. (1) Pertaining to a process in which all events occur one after the other; for example, serial transmission of the bits of a character according to V24 CCITT protocol. (2) Pertaining to the sequential or consecutive occurrence of two or more related activities in a single device or channel. (3) Pertaining to the sequential processing of the individual parts of a whole, such as the bits of a character or the

characters of a word, using the same facilities for successive parts. (4) Contrast with *parallel*.

server. A device, program, or code module on a network dedicated to a specific function.

server address space. Any address space that helps process work requests.

service. The amount of service consumed by an address space is computed by the formula:
service = (*CPU x CPU Service Units*)
+ (*SRB x SRB Service Units*)
+ (*IOC x I/O Service Units*)
+ (*MSO x Storage Service Units*) where CPU, IOC, MSO, and SRB are installation defined service definition coefficients. See also *service unit*, *CPU service unit*, *I/O service unit* and *storage service unit*.

service class. A subset of a workload having the same service goals or performance objectives, resource requirements, or availability requirements. For workload management, you assign a service goal and optionally a resource group to a service class.

service definition. An explicit definition of the workloads and processing capacity in an installation. A service definition includes workloads, service classes, systems, resource groups, service policies, and classification rules.

service definition coefficient. A value that specifies which type of resource consumption should be emphasized in the calculation of service rate. The types of resource consumption are CPU, IOC, MSO, and SRB.

The service definition coefficients are used to assign additional weight to one type of service relative to another, allowing you to specify which type of resource consumption should be emphasized in the calculation of service rates. For example, an IPS may contain the following service definition coefficients: `CPU=10.0 IOC=5.0 MSO=3.0 SRB=10.0` In this case, if an address space has accumulated 100 CPU service units, 200 I/O service units and 300 storage service units, and 10 SRB service units, its total accumulated service would be:

`(10x100)+(5x200)+(3x300)+(10x10)=3000 service units`

See also *CPU service unit*, *SRB service unit*, *I/O service unit* and *storage service unit*.

service level agreement (SLA). An agreement between the system administration group and a user group defining what service levels the former will provide to ensure that users receive the space, availability, performance, and security they need.

Service Level Reporter (SLR). An IBM-licensed program that provides the user with a coordinated set of tools and techniques and consistent information to help manage the data processing installation. For example, SLR extracts information from SMF, IMS, and CICS logs, formats selected information into tabular or graphic reports, and gives assistance in maintaining database tables.

service policy. A named set of performance goals, and optionally, processing capacity boundaries workload management uses this as a guideline to match resources to work. See also *active service policy*.

service request block (SRB) service units. A measure of the SRB execution time for both local and global SRBs, multiplied by an SRM constant which is CPU model-dependent.

service unit. The amount of service consumed by a work request as calculated by service definition coefficients and CPU, SRB, I/O, and storage service units. See also *CPU service unit*, *SRB service unit*, *I/O service unit*, *storage service unit* and *service definition coefficients*.

shared. Pertaining to the availability of a resource to more than one use at the same time.

side. A part of a partionable CPC that can run as a physical partition and is typically referred to as the A-side or the B-side.

single-image (SI) mode. A mode of operation for a multiprocessor (MP) system that allows it to function as one CPC. By definition, a uniprocessor (UP) operates in single-image mode. Contrast with *physically partitioned (PP) configuration*.

single mode optical fiber. An optical fiber in which only the lowest-order bound mode (which can consist of a pair of orthogonally polarized fields) can propagate at the wavelength of interest. Contrast with *multimode optical fiber*.

single-OS/390 environment. An environment that supports one OS/390 image. See also *OS/390 image*.

single point of control. The characteristic a sysplex displays when you can accomplish a given set of tasks from a single workstation, even if you need multiple IBM and vendor products to accomplish that particular set of tasks.

single system image (SSI). The characteristic a product displays when multiple images of the product can be viewed and managed as one image.

single-system sysplex. A sysplex in which only one system is allowed to be initialized as part of the sysplex. In a single-system sysplex, XCF provides XCF services on the system but does not provide signalling services between OS/390 systems.

skew. In capacity planning, the unsymmetrical distribution of I/O across BCUs on a system. Skew across BCUs is a function of the number of

controllers. That is, fewer controllers create a steep skew; more controllers create a less steep skew.

SLA. See *service level agreement*.

SLR. See *service level reporter*.

SPSSZR. The CP2000 S/390 Parallel Sysplex Quick-Sizer (SPS/SZR) is a PC-based (OS/2) productivity tool intended to assess the overall hardware requirements when moving applications from one or more individual System 390 processor environments to a Parallel Sysplex processing environment.

SRB service unit. Service request block (SRB) execution time for both local and global SRBs, multiplied by an SRM constant which is CEC model-dependent. Included in the execution time is the time used by the address space while executing in cross-memory mode (that is, during either secondary addressing mode or a cross-memory call). This execution time is not counted for the address space that the target of the cross-memory reference. See also *service unit*, *I/O service unit*, *storage service unit* and *service definition coefficients*.

standard deviation. Is a measure of how widely values are dispersed from the average value. Most products, such as RMF, which calculate the standard deviation for a set of sampled data, use the sample method to calculate the standard deviation. This method assumes that only a part of the whole population is known.

$$StdDev_{SampleMethod} = \sqrt{\frac{n \times \sum x^2 - (\sum x)^2}{n \times (n-1)}}$$

storage. A unit into which recorded data can be entered, in which it can be retained and processed, and from which it can be retrieved.

storage management. Storage management is the process of providing systems that use a consistent set of procedures which support management policies and data access on any platform anywhere. Host data, network originating data, and workstation data will be safely backed up and effectively managed. See also *system management*.

storage service unit. Storage service units are calculated using the formula:

(*central page frames*)*x*(*CPU service units*)*x*1/50

1/50 is a scaling factor designed to bring the storage service component in line with the CPU component. Not included in the storage service unit calculation are the central storage page frames used by an address space while referencing the private virtual storage of another address space through a cross-memory function (that is, through secondary addressing or a cross-memory call). These frames are counted for the address space whose virtual storage is being referenced. See also *service unit*, *SRB service unit*, *CPU service unit* and *service definition coefficients*.

structure. A construct used by OS/390 to map and manage storage on a CF. *See cache structure*, *list structure*, and *lock structure*.

subsystem. A secondary or subordinate system, or programming support, usually capable of operating independently of or asynchronously with a controlling system.

supervisor. The control program that enables the execution of other computer programs and regulates the flow of work in a data processing system. It may be control the execution of SCPs as with LPAR or it may be an SCP itself. See also *system control program*.

symmetry. The characteristic of a sysplex where all systems, or certain subsets of the systems, have the same hardware and software configurations and share the same resources.

synchronous. (1) Pertaining to two or more processes that depend on the occurrences of a specific event such as common timing signal. (2) Occurring with a regular or predictable timing relationship.

SYSID. The name assigned to a system image. The system can be a single system image in the CEC (in which case, the CECID and the SYSID will be the same), or the logical partition in the CEC. In OS/390 terms, this is the SMF SYSID.

sysplex. A set of OS/390 systems communicating and cooperating with each other through certain multisystem hardware components and software services to process customer workloads. There is a distinction between a base sysplex and a Parallel Sysplex. See also *OS/390 system*, *base sysplex* and *Parallel Sysplex*.

sysplex couple data set. A couple data set that contains sysplex-wide data about systems, groups, and members that use XCF services. All systems in a sysplex must have connectivity to the sysplex couple data set.

sysplex data sharing. The ability of multiple IMS subsystems to share data across multiple images. sysplex data sharing differs from two-way data sharing in that the latter allows sharing across only two OS/390 images.

sysplex failure management. The OS/390 function that minimizes operator intervention after a failure occurs in the sysplex. The function uses installation defined policies to ensure continued operations of work defined as most important to the installation.

sysplex management. The functions of XCF that control the initialization, customization, operation, and tuning of systems in a sysplex.

system. In data processing, a collection of people, machines, and methods organized to accomplish a set of specific functions.

system configuration. A process that specifies the devices and programs that form a particular data processing system.

system control program (SCP). Programming that is fundamental to the operation of the system. SCPs include OS/390, VM, and VSE operating systems and any other programming that is used to operate and maintain the system. Synonymous with *operating system*.

system management. The process of monitoring, coordinating, and controlling resources within systems. Systems management is the discipline and corresponding tools for planning, coordinating, and operating a computing system in order to keep it up and running, smoothly and effectively. It is comprised of the following categories:

- Business management. See also *business management*.
- Change management. See also *change management*.
- Operations management. See also *operations management*.
- Performance management. See also *performance management*.
- Problem management. See also *problem management*.
- Storage management. See also *storage management*.

S/390 microprocessor cluster. A configuration that consists of CPCs and may have one or more CFs.

S/390 Parallel (Transaction) Sysplex (SPTS) (CP2000). SPTS is a set of hardware- and software-capable system images which share applications and data. Transactions can be routed among the system images. Sharing data is accomplished by a specialized system facility called a CF. See also *sysplex*, *base sysplex* and *Parallel Sysplex*.

T

task control block (TCB). A dispatchable unit of work in OS/390.

throughput. (1) A measure of the amount of work performed by a computer system over a given period of time, for example, number of jobs per day. (2) A measure of the amount of information transmitted over a network in a given period of time.

tightly coupled. Multiple CPs that share storage and are controlled by a single copy of OS/390. See also *loosely coupled* and *tightly coupled multiprocessor*.

tightly coupled multiprocessor. Any CPC with multiple CPs.

transaction. In an SNA network, an exchange between two programs that usually involves a specific set of initial input data that causes the execution of a specific task or job. Examples of transactions include the entry of a customer's deposit that results in the updating of the customer's balance, and the transfer of a message to one or more destination points.

U

uncaptured time. The CPU time that cannot be attributed to a specific user or workload, thus there is no way to determine who is using the uncaptured time.

uniprocessor (UP). A CPC that contains one CP and is not partitionable.

UP. Uniprocessor.

V

velocity. A service goal naming the rate at which you expect work to be processed for a given service class or a measure of the acceptable processor and storage delays while work is running.

W

white space. Coupling Facility storage set aside for rebuilding of structures from other CFs, in case of failure.

WLM. Workload Manager, part of the OS/390 base control program. OS/390 workload management provides a solution for managing workload distribution, workload balancing, and distributing resources to competing workloads. OS/390 workload management is the combined cooperation of various subsystems (CICS, IMS/ESA, JES, APPC, TSO/E, OS/390 UNIX Services) with the workload manager (WLM) component.

workload. A group of work to be tracked, managed and reported as a unit. Also, a group of service classes. See also *business element* and *dimensioning workload*.

workload management mode. The mode in which workload management manages system resources on an OS/390 image. Mode can be either *compatibility mode*, or *goal* mode.

X

XCF. Cross-system Coupling Facility.

List of Abbreviations

ADMGDF	ADM Graphics Data File (GDDM)
AOR	Application owning region (CICS)
API	Application Program Interface
ASYNC	asynchronous
ATM	Automatic Teller Machine
AWM	Alternate Wait Management (OS/390)
BBU	battery backup
BCP	basic control program
BCU	basic configuration unit
BDAM	Basic Direct Access Method
BE	business element (CP2000)
BP	buffer pool
BWAT	Batch Workload Analysis Tool
CA	continuous availability
CAA	Cache Analysis Aid
CAU	CICS Affinity Utility
CBAT	CMOS Batch Analysis Tool
CBW2	Commercial Batch Workload 2
CB84	Commercial Batch 84
CCW	channel command word
CEC	Central Electronics Complex
CECID	CEC identifier
CF	Coupling Facility
CFR	Coupling Facility Receiver (channel)
CGI	Common Gateway Interface
CI	control interval
CMOS	Complementary Metal Oxide Semiconductor
COL	column
CP	capacity clanning
CP	control program
CP	central processor
CPC	Central Processing Complex
CPS	Capacity Planning Services
CPSTOOLS	Capacity Planning Services Tools
CPU	Central Processing Unit
CP90	Capacity Planning 90
CP2000	Capacity Planning 2000
CP2KEXTR	Capacity Planning 2000 Extract
CRR	Cache RMF Reporter
CR	capture ratio
CRE	Consistent Read Explicit (RLS)
CS	central storage
CS	Cursor Stability (DB2)
CSA	Common Storage Area (OS/390)
CSI	Consolidated Software Inventory (SMP/E)
CSTOR	Control Storage (Central Storage for CF)
CTC	channel-to-channel
CTL	control
CUA	Common User Access (SAA)
DASD	Direct Access Storage Device
DB	database
DBCTL	Database Control Subsystem
DB2PM	DB2 Performance Monitor
DCAT	DASD Cache Analysis Tool
DCTI	device connect time
DD	data definition
DFSMS	Data Facility Storage Management Subsystem
DFW	DASD Fast Write
DLS	Device Level Selection
DLSE	Device Level Selection Enhanced
DML	Data Manipulation Language
DOS	Disk Operating System
DP	dispatching priority
DS	data sharing
DSG	data sharing group
DYN	dynamic
EC	engineering change
EDF	Enterprise Data File
EDFI	Enterprise Data File
EMEA	Europe/Middle East/Africa
EMIF	ESCON Multiple Image Facility
EN	End-Node (VTAM)
EPDM	Enterprise Performance Data Manager (OS/390)
E/S	Engineering/Scientific
ES	Expanded Storage
ES	Enterprise Systems
ESCON	Enterprise Systems Connection
ESTOR	Non-Control Storage (Expanded Storage for CF)
ETR	external throughput rate
EV	execution velocity (WLM)
EXEC	execution program
FF	full function (IMS)
FIG	figure
FLIH	first level interrupt handler
FOR	File-Owning Region (CICS)
FP	Fast Path (IMS)
FPC1	Floating Point Commercial 1 (LSPR)
GB	gigabyte
GBP	Group Buffer Pool (DB2)
GDF	Graphics Data File
GR	generic resource
GRS	Global Resource Serialization
GTF	Generalized Trace Facility
gid	UNIX group identity
GUI	graphical user interface
HFS	Hierarchical File System (UNIX)
HONE	hands-on network environment
HORIZ	horizontal
HPOOL	hiperpool
HPTS	Highly Parallel Transaction Server
HSA	Hardware Service Area
HST	Hiperspace Service Time
IIT	I/O interrupt time
I/O	input/output
IOP	input/output processor
IOS	input/output supervisor
IBM	International Business Machines Corporation
ICMF	Integrated Coupling Migration Facility
ICS	Installation Control Specification (OS/390)
ics	internet connection server

© Copyright IBM Corp. 1998

ID Identifier
IML initial microcode load
IMS Information Management System
IPL initial program load
IPS Installation Performance Specification (OS/390)
IRLM Integrated Resource Lock Manager
IRWW IBM Relational Warehouse Workload (DB2)
ISPF Interactive System Productivity Facility
ITR internal throughput rate
ITRR internal throughput rate ratio
ITSC International Technical Support Center
ITSO International Technical Support Organization
JCL job control language (JES)
JES Job Entry Subsystem
KB kilo byte
KGTV Korhonen George Thorsen Vaupel
KLOC kilo lines of code
Km kilometer
LAN local area network
LCU Logical Control Unit
LDR law of diminishing return
LP logical partition
LPAR logically partitioned mode
LRU least recently used
LSPR Large Systems Performance Reference
LU logical unit
LUE low utilization effect
M many (tasks) (CP2000)
MA migration age
MB mega byte
Mbps mega bits per second
MBps mega bytes per second
MGR manager
MHz megahertz
MIPS millions instructions per second
MLPF Multiple Logical Processor Facility (Hitachi Data Systems Corp.)
MM multimode
MP multi-processor
MPL multi-programming level
MPP message processing program (IMS)
MPU multi-programming level
MRO multi-region operation
MVPG move page
MVS Multiple Virtual Storage
NC no commit (DB2)
NN Network-Node (VTAM)
NR Network-Node (VTAM)
NRI no read integrity (RLS)
OEM original equipment manufacturer
OLTP Online Transaction Program
OMVS OpenEdition MVS (OS/390 UNIX Services)
OS operating system
OS/390 Operating System/390
OSAM Overflow Sequential Access Method
PER program event recording
PAR peak-to-average ratio
PCR processor capacity reference
PDS partitioned data set
PDSE partitioned data set enhanced
PES S/390 Parallel Enterprise Server
PF program function
PGM program
PGN performance group number
pid processid (UNIX)
PLPA pageable link pack area (OS/390)
PM performance management
PM Presentation Manager
PMIO Performance Management Input/Output
PMOS Performance Management Offerings and Services
PP physical partitioned (mode of operation)
PR/SM Processor Resource/System Manager
PROFS Professional Office System
PRTY priority
PTF program temporary fix
PTS S/390 Parallel Transaction Server
PWS Programmable Work Station
QOR Queue-Owning Region (CICS)
RACF Resource Access Control Facility
RAS reliability availability serviceability
RCT region control task
RECON recovery control
RIOC relative I/O content
RIPS relative instructions per second
RISC Reduces Instruction Set Computing
RLCK relative lock content
RLS record level sharing
RMF Resource Measurement Facility
RMFGAT RMF gatherer
RMWT residual mean wait time
ROT rule of thumb
RPGN reporting performance group number
RPP relative processor performance
RR repeatable read (DB2)
RS read stability (DB2)
RT response time
SAP service assist processor
SCA shared communication area (DB2)
SCP system control program
SDP saturation design point
SEL select
SEMHQ Shared Expedited Message Handling Queue (IMS)
SESS session
SEQ sequential
SHR shared
SI single image (mode of operation)
SIO start I/O
SLA service level agreement
SLIH second level interrupt handler
SLIP serviceability level indicator processing
SLR Service Level Reporter
SM single mode
SM single (task) multi (thread) (CP2000)
SMF Systems Management Facility
SMQ Shared Message Queue (IMS)
SMS Systems-Managed Storage
SNAP/SHOT System Network Analysis Program/ Simulation Host Overview Technique
SP System Product

SPEC	system performance evaluation cooperative
SPOOL	simultaneous peripheral operation on-Line
SPTS	S/390 Parallel Transaction Server
SPTS	S/390 Parallel Transaction System (CP2000)
SQA	system queue area (OS/390)
SQL	Sequential Query Language
SRB	service request block (MVS control block)
SRM	System Resources Manager
SS	single (task) single (thread) (CP2000)
SSI	single system image
ST	service time
SU	service unit
SUBSYS	subsystem
SVS	Solutions Validation Services
SW	software
SYN	synchronous
SYNC	synchronous
SYSID	system identifier
SYSPLEX	systems complex
TCB	task control block (OS/390)
TEMPSTOR	temporary storage (CICS)
TM	Transaction Manager
TOR	Terminal-Owning Region
TPC	Transaction Processing Performance Council
tpm	transactions per minute
TPNS	Teleprocessing Network Simulator
TSO	time sharing option
TXT	text
UIC	unreferenced interval count
uid	UNIX user identity
UP	uni-processor
UPS	uninteruptible power supply
UR	uncommitted read (DB2)
URL	Universal Resource Locator
USA	United States of America
VLF	Virtual Lookaside Facility
VM	Virtual Machine
VMPRF	Virtual Machine performance
VOLSER	volume serial
VSAM	Virtual Storage Access Method
VTAM	Virtual Telecommunications Access Method
WLM	Workload Manager
WSC	Washington System Center
XCF	Cross-System Coupling Facility
XES	Cross-System extended services
XI	cross-invalidate
XRF	Extended Recovery Facility

Index

Numerics

321-based processors
 See ES/9000
340-based processors
 See ES/9000
511-based processors
 See ES/9000
520-based processors
 See ES/9000
711-based processors
 See ES/9000
9021 55

A

abbreviations 313, 327
acronyms 313, 327
analytic method 5
 ANCICS 5
 BEST/1 5
 CP2000 5
 SLR/CP 5
ANCICS 5
Annex 10
AOR
 See application-owning region
APAR
 CF considerations for shared LPs 31
 II09146 35
 II09294 31
APPC
 creating address spaces 110
application
 buying new 43
 capture ratio 16
 CICS migration to other platforms 274
 client 284
 commit 65
 data sharing part of 66
 evaluating for Parallel Sysplex 7
 IMS/DB data sharing example 77
 moving to Parallel Sysplex 199
 priority 216
 production volume stress testing 7
 RIOC 32
 single-threaded 192, 224
 sysplex migration 57
 VSAM RLS data sharing example 89
application capture ratio 16
application-owning region
 TOR routing overhead 63
APPN
 SNAP/SHOT modelling 284
architecture
 Parallel Sysplex 45
 S/370 9
 S/390 9
ASKQ
 WSC 268
aspects 267
 of capacity planning 267
 Parallel Sysplex 267
availability
 and capacity planning 41
 CF redundancy 201
 CF structures 57
 characteristics of DB2 CF structures 64
 considerations for Parallel Sysplex 58
 contingency allowances 41
 continuous 46
 DB2 recommendation 64
 improved for DB2 data 64
 recommendation for number of CFs 173

B

backup
 capacity 22
 CF 57
balanced system 20, 39, 42
batch
 and high locking rates 79
 and M/M/c assumptions 223
 avoid long-running concurrent with online 53
 Batchpipe/MVS 19
 BWAT 28
 BWATOOL package 306
 CB84 11, 251
 CBAT 274
 CBW2 11
 combining with interactive 5
 DB2 batch unlock 211
 DFSORT hipersort 37
 engineering/scientific 19
 hiperbatch 37
 long-running 19, 33
 short-running 19, 33
 the effect of many small steps 19
 unlock rate 188
benchmark 5
 accuracy 5, 6
 EMEA 7
 IBM-conducted 9
 LSPR 9
 PCR 5
 PSSC 7
 requirements 5
 services 7

© Copyright IBM Corp. 1998

benchmark *(continued)*
SPEC 6
SVS 7
TPC-C 6
TPNS 5
types of 5
BEST/1 5
bibliography 305
BLSRAID 272
bottlenecks 20
buffer
DB2 32
effect of 5
HSB 10, 29
invalidation 33, 53
local coherency 62
OSAM 75
SMQ 62
VMA 284
VSAM 75
buffer pool 49
cache structure 93
coherency 50
coherency considerations 49, 50
CP2000 example 206
CP2000 sizing 71
DB2 ALTER TABLESPACE command 71, 72
DB2 cache structure 64
DB2 GBP 61, 65
DB2 sizing 63
DB2 sizing summary 73
DB2 virtual 70
factors influencing DB2 sizing 71
global directory 51
hiperpools 70
IMS buffer pools 62
IMS/DB 80
Local Pool structure 93
Parallel Sysplex 35
RLS 91
VSAM 37
business elements 23, 41
BWATOOL package 272, 306

C

CA 274
CA17 274
cache structure
an example 51
DB2 GBP 64
directory only 75
OSAM buffers 75
VSAM buffers 75
capacity planning 39, 43, 267
activities 39
and I/O 31
aspects of 267
balanced system 3, 42
capacity planning *(continued)*
balanced system concept 20
benefits 3
CF 3
CFCC 30
collect and document results 39
collecting and documenting 43
communicating results 44
Coupling Facility 4
CP2000 162
CPU 3, 8
DB2 access rate modelling 184
deciding method 40
different levels of 272
dimensioning workload 41
effects of workload balancing 48
engine size issues 192
factors influencing 272
future predictions 42
how often 44
I/O 3
I/O projections 194
ICMF 30
introduction 2
level of detail 273
LPAR 30
M-value table 293
methods 4, 273
MIPS 8
model and interpret data 39, 43
Parallel Sysplex 3
positioning 7
program products 272
questions 2
reality-check 21
roadmap 163
services 272
storage 3, 35
storage projections 192
time invested 273
tools 267, 271
tools catalogue 271
why 2
workload and CPU projections 191
workload balancing effects 42
capacity planning method 40
capture ratio 16, 17
application 16
causes of low 17
CPU TCB time 14
CPU wait time 14
definition 14
DIM 19
factors influencing 17
how to assess 16
how to prorate non-captured time 14
LPAR management time 17
Parallel Sysplex 20

capture ratio *(continued)*
RMF 14
SRB time 14
system 16
typical values 19
case study: CF modelling
answers 238
CEC request activity 228
CF input data 231
environment 226
input 228
link input data 232
questions 226
quick graph 238
results 236, 239
spreadsheet 227
using the model 233
case study: DB2 migration
display CF window 213
further statistics required from DB2PM 207
future locking rate calculation 205
locking rate calculation 204
migration to a data sharing environment 197
output window 212
processor utilization graph 214
Quick-Sizer Default 198
Quick-Sizer input window 198
Quick-Sizer optional input window 200
Quick-Sizer output window 202
total CF access rate and split 210
case study: processor speed
analyze processor speed 215
CP2000 actions 216
description 216
processor response time 221, 222
processor service time 220
processor utilization 219
single CP analysis 221
WLM goal mode 225
Catalogue
capacity planning 271
CB84 11, 251, 254, 255
CBAT 274
CMOS batch analysis 274
CBW2 11
CD13 275
CEC
capacity planning 3
CPENABLE recommendation 31
LPAR/CE 30
LSPR 9
MIPS 8
MP factor 29
out-of-gas 2
Parallel Sysplex support 46
participating in Parallel Sysplex 54
shared or dedicated CPs 30
stressed 2
CEC *(continued)*
tightly coupled architecture 29
CECs 6
OEM 6
Central Processor (CP)
CF LP 30
CPENABLE recommendation 31
CPU time definition 8
engine speed 192
light load 23
logical defined per real CP 256
LP weight 31
LPAR/CE 249
M-value table 293
management 31
maximum load 24
MHz 8
MIPS 8
overload 25
resources needed for LPs 254
response time 26
shared or dedicated CPs 30
simple rules of thumb 4
SOFTCAP tool 283
speed of CEC 65
speed of CF 65
wait 14
central storage 18
lack of 18
CF
access service times 131
activity spreadsheet 133
availability 58
backup 57
capacity planning 3
capacity planning model 144
CFCC capacity planning 30
Code Levels 56
considerations 61
CPU Speed 59
DB2 access rate modelling 184
DB2 batch unlock 211
DB2 structure usage 64
Dynamic dispatching capability 56
false contention 54
function 55
hot standby 58
ICMF assist 30
ICMF capacity planning 30
IMS access rate modelling 188
invalidates 52
link service 151
modelling 61
one-way versus N-way 58
queue time calculations 151
queueing formulas 149
real contention 54
redundancy 201

CF *(continued)*
request model 148
request processing 144
requests and delay types 127, 147
RMF reports 119, 157
service time calculation 152
shared or dedicated CPs 30
storage 58
structures 56
time distributions 154
times, delays and queues 147
total contention 54
understanding CF processing 126
using the CF processing model 158
utilization 58
CF channel
See Coupling link
CF link
See also Coupling link
HiPerLinks 60
CFCC 55, 256
See also Coupling facility control code
channels
capacity planning 3
CP2000 specification 173, 201
non-ICMF 31
checkpoint
frequency 53
frequent 53
how to obtain I/O number 190
JES2 57, 67
shared 67
CICS 11
CICS and CICS TS
CF sizing 100
DFHLOG 93
DFHLOG Sizing 96
DFHSHUNT 94
DFHSHUNT Sizing 97
forward recovery logs sizing 97
ITR considerations 100
log of logs 94
logging considerations 93
M-value 11
monitoring the logger 98
RIOC 32
sizing utility 95
structure 57
temporary storage 57, 98
user journal 94
user journal sizing 97
VSAM RLS 84
VSAM RLS integrity 84
VSAM RLS recoverable/non-recoverable 85
XCF MRO 62
CICS processor utilization tool 275
CMOS
batch analysis 274
CMOS *(continued)*
processor utilization tool 275
collision analysis 25
commands
CPSTOOLS 278
DB2 ALTER TABLESPACE 71, 72
constraints 20
consultant MIPS 6
contention
capture ratio and link contention 20
false 53, 54
false CF contention 54
how much is acceptable 54
how to decrease for locks 53
how to reduce 53
lock rates 78, 89
real 53
real CF contention 54
recommendation for 54
recommendation for false contention 54
recommendation for real contention 54
recommendation for total global contention 54
RLKC 31
RMF reporting 54
total 54
total CF contention 54
tuning workload to avoid CF contention 53
contingency allowance 42
continuous availability
demand for 49
cost
allocation 19
data sharing 15
DB2 Parallel Sysplex CPU 65
DB2 Parallel Sysplex CPU factors 65
difference between ES and CS storage 36
factors in Parallel Sysplex 62
IMS Parallel Sysplex CPU 74
LPAR 254
multisystems management 66
of capacity planning 4
of managing n-way data sharing in non-sysplex environment 50
of queueing 201
Parallel Sysplex 15
Parallel Sysplex CPU 63, 86
Parallel Sysplex enablement 66, 77, 88
Parallel Sysplex incremental growth 66, 77, 88
Parallel Sysplex multisystems management 77, 88
Parallel Sysplex total 67
Coupling Facility
See CF
Coupling Facility channel
See Coupling link
Coupling facility control code
coupling links 256
CP2000 modelling 189
ICMF alternative 30

Coupling facility control code *(continued)*
in stand-alone 9674 256
level 65
together with MVS 256
Coupling Facility link
See Coupling link
Coupling Facility structure
See cache structure
See list structure
See lock structure
Coupling link
data transfer time 131
distance effect 60
effect of distance 60
HiPerLinks 60
CP2000 5
buffer pool sizing 71
capacity planning 162
collision analysis 25
CP2KEXTR 306
CP90DOC 306
group buffer pool sizing 71
GRS 188
how to provide input 287
I/O projections 194
IMS buffer pool sizing 74
M-value 11, 13
M-value table 293
outline of usage 162
overview 162
PCR 306
positioning 272
Quick-Sizer 4
Quick-Sizer overview 164
RLKC 34
specifying channels 173, 201
storage projections 192
XCF 57, 62
CP2KEXTR package 306
CPAIP
CPOS 272
CPR 277
LSPR/PC 277
CPSTOOLS 278
CPU 267
processor storage 267
storage 267
CPU time
application capture ratio 16
capacity planning 8
capture ratio 14
capture ratio definition 15
capture ratio values 19
captured time 14
cost for IMS/DB data sharing 77
cost for RLS data sharing 88
CP2000 projections 191
CPU service units 29
CPU time *(continued)*
cycle time 8
high service time 18
HSB 10, 29
IMS/DB locking rates 78
M-value 8
MIPS 8
non-captured time 14
Parallel Sysplex capture ratios 20
Peak-to-Average Ratio 21
queue time 27
response time 27, 28
response time graph 26
RPP 8
Saturation Design Point 22
SDP considerations 25
service time 27
service units 29, 33, 38
System Capture Ratio 16
time measurement 9
CQS 83
cross-invalidate 52
Cross-system Coupling Facility (XCF)
charging the service time 65
CICS MRO 62
CP2000 57
RMF activity report 190, 289
services 20
SMF 74.2 record 288
SNAP/SHOT simulation 284
storage 35
structure 57
usage by member 190
Cross-system extended services
charging the service time 65
IRLM number of call 190
multisystem locking 34
overhead 33
services 20
CS/ES considerations 285
CTC
for 2-way IMS/DB data sharing 50
IRLM 50
VTAM 50
cycle time 8

D

DASD 267
data integrity
in a single OS/390 (MVS) image 49
in multiple OS/390 (MVS) images with Parallel Sysplex 51
in multiple OS/390 (MVS) images without Parallel Sysplex 49
in the Parallel Sysplex 51
structures 51
data sharing
cost of 15

data sharing *(continued)*
cost of managing 50
DB2 51
IMS/DB 51
data sharing group 64
database manager
See DB2
See IMS
DB2 11
See also database manager
a data sharing example 51
ALTER TABLESPACE command 71, 72
availability recommendation 64
batch unlock 211
buffer 32
buffer pool sizing 63, 70
buffer pool sizing summary 73
buffer pools 61
cache structure 64
CBW2 11
CF access rate modelling 184
CF structure usage 64
cpu cost for data sharing 65
data sharing considerations 63
data sharing group 64
DB2PM 68, 185
GBP 61
GBP structure sizing 72
GBPCACHE ALL 64
GBPCACHE CHANGED 64
hiperpools 70
hot spots 73
improving data availability 64
list structure 64
local buffer pool report 185
lock granularity 53
lock reductions 68
lock structure 64
lock structure sizing 71
locking rate 67
multimode effect on ITR 59
no I/O growth 64
Parallel Sysplex RIOC 33
performance and capacity planning 63
R/W ratio 68
RIOC 32
RLKC 34
SCA structure sizing 71
store-in technique 61
structure 57
suggested m values 69
Sysplex data sharing 51
virtual buffer pools 70
workload balancing 64
DB2PM 68, 185
DBDGEN 81
DCAT 279
DCATAIDS package 279
DFSMS/MVS
DFSORT 37
See also sort
DIM 19
dimensioning workload 22, 41
discrete method 5
SNAP/SHOT 5
disk
See DASD
distance
effect on coupling links 60
DL/1
See IMS
DMAGIC/2 280
DMAGIC2 package 280
dump
SMFDUMP 288
dynamic CF dispatching 56

E

EHONE
EPDM 5, 272
EPDM/CP 272
Erlang
M/M/c 26
response time 26
ES/9000
Parallel Sysplex support 46
PR/SM planning guide 31, 306
sharing CPs 30
SNAP/SHOT 284

F

false contention 54
See also contention
fiber
See channels
See Coupling link
FLIH 14
fluctuation 42
fork 110
formula
See Rules of thumb
future predictions 39, 42

G

Gartner Group 10
GBP
See buffer pool
See DB2
glossary 313
GRS
cost included in multisystems management cost 67
CP2000 188

GRS *(continued)*
reserve/release 49
XCF usage by member 190
guidelines 4

H

Hardware Configuration Definition (HCD)
hashing 53, 54
HBAID 272
HCD
See Hardware Configuration Definition (HCD)
Hiperbatch 37
Hipersort 37
HONE
See EHONE
HSA
vector bits 51
HSB 10, 29
HSB misses 29

I

I/O
buffer invalidation 33
capacity planning 3
CPENABLE recommendation 31
growth in IMS/DB data sharing environment 76
projections 194
relative content 31
RIOC 31
service units 29, 33, 38
VIO 37
ICMF 56, 256
See also Integrated Coupling Migration Facility (ICMF)
IDC 10
IMS 11
See also database manager
2-way data sharing 50
application data sharing example 77
batch considerations 79
buffer pool sizing 74, 80
buffer pools 62
cache structures 75
CF access rate modelling 188
CPU cost for data sharing 77
data sharing group 76
data sharing publications 75
DBDGEN 81
expected I/O growth 76
Expedited Message Handler 76
Fast Path Databases 76
IMS/ESA V6 VSO DEDB block level sharing 76
IRLM 50
IRLM Lock Structure 75
lock granularity 53
lock traces 78, 79
locking rates 78
IMS *(continued)*
M-value 11
OSAM and VSAM structures 81
OSAM buffers 75
Parallel Sysplex CPU cost 74
Parallel Sysplex data sharing 74
Parallel Sysplex RIOC 33
PTSLOCK package 306
RIOC 32
RLKC 34
Shared Message Queue 76
store-in technique 62
store-through technique 62
structure 57
V5 block level sharing 75
V6 block level sharing 75
virtual fetch 37
VSAM buffers 75
VTAM CTCs 50
IMS/DB
See IMS
Integrated Coupling Migration Facility (ICMF)
assist 30
CFCC alternative 30
Internal Coupling Facility (ICF) 56
International Centers
ITSO 272
IRLM
2-way data sharing 50
lock structure sizing 71
structure 57
VTAM CTC 50
ISC link
See Coupling link
ITR
RPP 11
ITSO 272

J

JCL
RMF Postprocessor sample 288
JES2
checkpoint 57, 67
checkpoint data set number of I/Os 190
structure 57
JES3
CBAT 274
JESXCF 35

K

KGTV 152, 226, 269, 281
roadmap 162
KGTV package 281

L

Large Systems Performance Reference (LSPR)
 CB84 11
 CBW2 11
 CICS 11
 DB2 11
 IMS 11
 introduction 10
 ITRR 11
 RPP 11
 TSO 11
latent demand 20, 21, 23
 examples of 23
level of detail 273
light-weight 111
linear projection 4
link
 See Coupling link
list structure
 allocation 57
 CICS 57
 CICS log 58
 CICS/VSAM RLS 57
 DB2 64
 generic resources 57
 JES2 57
 logrec 58
 multi-node persistent session (MPNS) 57
 parallel batch 58
 SCA sizing 71
 SmartBatch 58
 syslog 58
 system logger 58
 tape 57
 VTAM 57
 XCF 57
lock structure
 CICS/VSAM RLS 57
 DB2 57, 64
 expedite message handler (EMH) 57
 false contention 53, 54
 granularity 53
 GRS resource serialization 58
 GRS Star 58
 hashing 53
 IMS 57
 IRLM 57
 shared message queue (SMQ) 57
 sizing 71
 synonyms 53
 virtual storage option (VSO) data entry database (DEDB) 57
LP
 See LPAR
LPAR
 capacity planning 30
 cost 254
 CPENABLE recommendation 31
LPAR *(continued)*
 ICMF assist 30
 LPAR/CE 30
 LPARCE 281, 306
 management time 17
LPAR/CE 30, 281
 CFCC using real coupling links 256
 CFCC using the ICMF facility 257
 implementation and operation 253
 input 253
 intention of 253
 introduction 253
 output 254
 Parallel Sysplex considerations 256
 partition capacity 255
 real cp resource usage 255
LPARCE 281
LPARCE package 253, 281, 306
LSPR 9, 281
 IBM benchmarks 281
LSPR/PC 251, 282
 CP2000 PCR 282
 functional overview 252
LSPRPC 282
LSPRPC package 252, 282, 306
LTERMs 83

M

M-value 8
 CICS 11
 CP2000 11
 definition 11
 how to obtain 11
 IMS 11
 RIOC 31
 TSO 11
Markow theory 27
MAS
 See Multi-Access Spool
measure current workload 39
measuring current workload 41
medium-weight 111
message passing 49
Meta Group 11
methods 4
 analytic method 5
 capacity planning 273
 CPU time measurement 9
 discrete method 5
 guidelines 4
 linear projection 4
 steps 273
MFTS package 78, 89, 306
migration
 See Integrated Coupling Migration Facility (ICMF)
MIPS 8
MKTTOOLS 272

MM
See Coupling link
MP factor 29
Multi-Access Spool
Multimode
See Coupling link
MVS
data integrity 49
OS390PRF package 306
OS390T97 package 306

O

OEM CECs 6
OPC/ESA
OS/390 UNIX Services Sizer Tool
how to use 240
input details 242
OS/390 results 243
primary panel 241
sizing computation 243
OS390PRF package 306

P

package
BV390 306
BWATOOL 306
CA17 274
CD13 275
CP2KEXTR 306
CPR 277
DCATAIDS 279
DMAGIC2 280
GF225009 306
KGTV 152, 226, 269, 281
LPARCE 253, 281, 306
LSPRPC 252, 282, 306
MFTS 78, 89, 306
OS390PRF 306
OS390T97 306
PCR 306
PTSLOCK 79, 306
SOFTCAP 283, 306
SPSSZR 306
SWPRICER 306
paging 37
PAR 21, 23
examples of 23
Parallel Sysplex 256, 257, 267
buffer pool 35
business value 306
capacity planning 3
capture ratio 20
CECs participating 54
CFCC using real coupling links 256
CFCC using the ICMF facility 257
continuous availability 46
cost factors 62
Parallel Sysplex *(continued)*
CPU 267
data integrity 49, 51
HW components 54
IMS buffer pool sizing 74
IMS/DB data sharing 74
initial cost 67
investment preserved 46
latent demand 20
locking 53
Quick-Sizer Parallel Sysplex migration 181
RIOC 33
scalability 46, 47
total cost 47
workload balancing 21, 48
PCR 5
basic analysis 250
how to start 250
introduction and background 249
PCR and LSPR/PC 249
PCR package 277, 306
PDSE 37
peak 41
peak-to-average ratio 21
performance
CF access service times 131
keeping CF contention low 53
LPAR considerations 30
performance group number 41
performance management 272
Performance Management Services 262
performance management CICS 263
performance management DB2 264
performance management I/O 264
performance management MVS 262
PGN 41
performance group number 41
PMCICS 262
PMDB2 262
PMIO 262
PMMVS 262
positioning 267
CP2000 272
Tools 267
PR/SM 30
predict the future 39
predicting the future 42
processor
See Central Processor (CP)
See CF
See CMOS
See ES/9000
Processor Capacity Reference 249
processor utilization tool 275
production
CF LP considerations 30
external CF links 56
LSPR workloads 9

production *(continued)*
number of concurrent forks (OS/390 UNIX Services) 115
volume stress testing 7
Program Products
capacity planning 272
PTSLOCK package 79, 306

Q

queueing theory
and capacity planning 5
and CF processing 227
concepts 297
KGTV tool 227
literature 1, 143
with computer science applications 143
Quick-Sizer 4
access rate input for others 178
accuracy 165
after migration 183
before migration 181
CF storage modelling 189
CFCC 189
DB2 access rate modelling 184
defaults 165, 178
defining your environment 165
description 278
engine size issues 192
how to obtain 278
IMS access rate modelling 188
input window 166
optional input window 169
output window 171
overview 164
Parallel Sysplex migration 181
sample scenario 165
service time analysis 175
structure 165
workload and CPU projections 191

R

RACF
structure 57
ratio of peak-to-average 21
formula 21
real contention 54
See also contention
recommendation
for total locking contention 54
record level sharing
application data sharing example 89
Buffer Pool sizing 93
Buffer Pool Sizing ROT 93
cache sizing 93
cache sizing ROT 93
cache usage and sizing 91
CICS/VSAM data sharing 84
record level sharing *(continued)*
CPU cost for data sharing 88
data sharing 87
I/O due to buffer invalidation 33
I/O growth 87
list structure 57
lock structure 57
locking calculations 89
MRO considerations 86
multimode effect on ITR 59
structures 57
VSAM 57
relative I/O Content 31
relative lock content
DB2 34
definition 31, 34
IMS 34
relative processing power (RPP)
ITR 11
when not to use 11
workload composition 11
reporting performance group number 41
response time 27
RIOC 31
RIOC in Parallel Sysplex 33
RIPS 8
RLS
See record level sharing
RLS data sharing 87
RMF 118
available spreadsheet macros 260
capture ratio 14
CF activity report 123, 190
CF activity spreadsheet 133
CF cache structures 120
CF Data Collection 118
CF processor summary 122
CF storage summary 122
CF structure activity 124
CF structure summary 120
CF subchannel activity 126
CF usage summary 120
channel activity report 137
Coupling Facility reports 119
Data Buffer 119
data collection 118
how to read the CF activity report 131
list structures 121
lock structures 121
Monitor III 118
OMVS kernel activity report 141
reporting 118
SMF data buffer storage 35
spreadsheet converter 260
spreadsheet reporter 259, 260
Sysplex Data Server 119
Sysplex-wide Reports 118
typical CF access service times 131

RMF *(continued)*
XCF activity report 190, 289
XCF activity reports 138
RPGN 41
reporting performance group number 41
RPP
See relative processing power (RPP)
RPQ
to extend distance of multimode link 168
to extend distance of Multimode links 59
Rules of thumb
See also Rules of thumb
central and expanded storage 35

S

S=M x R x B 31
saturation design point 22
scalability 46
SDP 22, 23
examples of 23
service level agreements 42
service time 27
sharing
CICS temporary storage 57
CPU cost for IMS/DB 77
CPU cost for VSAM RLS 88
data integrity 49
expected I/O growth for IMS/DB 76
IMS/DB 74
Single mode
See Coupling link
skew 21
SLA 42
SLR/CP 5
SM
See Coupling link
SMF
74.2 record (XCF) 288
RMF data buffer 35
SMFDUMP 288
SNAP/SHOT 5, 284
SOFTCAP 283
SOFTCAP package 283, 306
effects of software migration 283
SNAP/SHOT 284
sort
See DFSORT
spawn 110
SPSSZR package 306
SRB time 14
SRM
CPENABLE recommendation 31
CPU service units 29, 33, 38
IOC service units 29, 33, 38
MSO service units 29, 33, 38
SRB service units 29, 33, 38
Steps
capacity planning 273
storage 267
capacity planning 3, 35
central 35
central or expanded 35
central versus expanded 36
CF usage 57
console 35
cost difference between ES and CS 36
DASD 267
DFSORT hipersort 37
hiperbatch 37
IMS virtual fetch 37
paging 37
PDSE 37
projections 192
service units 29, 33, 38
SMF data buffer 35
SOFTCAP tool 283
specific need for expanded storage 37
swapping 37
VIO 37
VSAM hiperspaces 37
XCF 35
structure
See also cache structure
See also list structure
See also lock structure
allocation 57
CICS 57
CICS log 58
CICS/VSAM RLS 57
DB2 57
GBP structure sizing 72
granularity 53
GRS resource serialization 58
GRS Star 58
IMS 57
IRLM 57
JES2 57
lock structure sizing 71
logrec 58
parallel batch 58
RACF 57
SCA structure sizing 71
SmartBatch 58
storage 57
syslog 58
system logger 58
tape 57
VSAM 57
VTAM 57
XCF 57
SVS 7
swapping 37
SWPRICER package 306
synonyms 53
system capture ratio 16

system outage 41
systems management
See DFSMS/MVS
See SMF

T

tape
structure 57
TCP/IP
threads 111
time 273
invested in capacity planning 273
tools 267
See also package
BWAT 28
CA17 274
capacity planning 267, 271
catalogue 271
CBAT 274
CICS/VSAM RLS capacity planning 278
CMOS Batch Analysis Tool 274
CMOS processor utilization tool 275
CP2000 (capacity planning 2000) 276
DB2 Parallel Sysplex input for CP2000 279
DCAT 279
DMAGIC/2 280
DMAGIC/2 (DASD and tape magic/2) 280
IMSAFAID (IMS affinity aid) 280
IMSSMQCP (IMS shared message queue capacity planner) 280
KGTV (a CF processing model) 152, 226, 269, 281
LPAR/CE 281
LSPR/PC 282
OS/390 UNIX Services sizer 283
overview 267
PCR 277
PMOS 262
positioning 267, 272
Quick-Sizer 278
RLS lock sizer 282
SOFTCAP 283
SPSSZR 278
VMA(Volume Mount Analyzer) 284
total contention 54
See also contention
transaction manager
See CICS and CICS TS
See IMS
See TSO
TSO 11
M-value 11
PDSE 37
RIOC 32
tuning
Parallel Sysplex configuration to keep false CF contention low 54
workload to avoid CF contention 53

U

UIC 18, 285
UNIX
application integration 107
comparison of benchmarks 106
creating new processes 110
file system usage 112
memory usage 114
methodology 102
OS/390 perspective 109
pathlength considerations 113
sizer tool 115
SPECint benchmarks 104
the bridge model 116
threads 111
TPC-A benchmark 105
TPC-C and reality 105
TPC-C benchmark 104
workload management 108

V

VIO 37
Virtual fetch 37
Virtual I/O 37
VMA 284
VSAM
hiperspaces 37
structure 57
VTAM
CF access rate 62
CTCs for 2-way IMS/DB data sharing 50
part of data sharing RPP 14, 19, 66, 77, 88, 167, 199
SOFTCAP 283
structure 57

W

WLM 41
See also Workload Manager
workload 23, 41
BWATOOL package 306
CB84 11
CBW2 11
CICS 11
collision analysis 25
combining batch and interactive 5
CP2000 projections 191
DB2 11
dimensioning 22, 41
dimensioning workload 22
distribution 23
engine size issues 192
examples of 23
fluctuation 42
IMS 11
latent demand 21

workload *(continued)*
M-value 11
manager 41
mix 251
peak 41
pent-up demand 21
TSO 11
workload balancing
DB2 64
effect on capacity planning 42
Parallel Sysplex 21
Workload Manager 41
workload measurement 39

X

XCF
See Cross-system Coupling Facility (XCF)
See Cross-system extended services
Xephon 11
XES
See Cross-system Coupling Facility (XCF)
See Cross-system extended services

Z

ZEKE 274

ITSO Redbook Evaluation

OS/390 MVS Parallel Sysplex Capacity Planning
SG24-4680-01

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at http://www.redbooks.com
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@vnet.ibm.com

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes____ No____

If no, please explain:

What other redbooks would you like to see published?

Comments/Suggestions: **(THANK YOU FOR YOUR FEEDBACK!)**

© Copyright IBM Corp. 1998



Printed in U.S.A.

